# PRODUCT SPECIFICATIONS

RENAULT

**36 - 02 - 014 / - - G**

**Rules for the management of**

**on-board Data Identifiers in UDS implementation.**

**Renault Automobile standardisation**
**DQ - EDA / Department 67250**

This document is to be considered as a whole, the parts of which shall not be separated.



**FIRST ISSUE**

| | | |
|---|---|---|
| January 2010 | 1.0 | First version |
| May 2010 | 2.0 | --Add Traceability format<br>-Add The identifier (DID) F1.87 |
| Jan 2011 | --A | -Simplification of rules: Only 3 important references in an ECU<br>-Length of DID F1.8A 120 bytes ➔ 64 bytes |
| Jan 2012 | --B | -Some Corrections<br>-Chapter B.1➔ Add order part number reference (F1.A1) for Nissans Needs.<br>-Chapter D ➔ Add F0.12 and explanations when to use it.<br>-Rule F.2 ➔ tracability Nissan code defined by ECU designer<br>-Rule G.1 (Table) ➔<br>Add information for better understanding<br>Add F1.80 DID for Engine ECU needs |
| June 2013 | --C | Remove Write one for VIN data<br>Explanation about F1.88 & F0.12 DID<br>Minor modifications |
| Sept 2013 | --D | -VIN write in extended session of application and not in boot session<br>-Deletion of Rule A.2 because confusing with Rule G.2 |
| Dec 2013 | --E | -Rewrite the document with unitary requirement with more precision |
| June 2014 | --F | **Major Modifications:**<br>• The VIN must be modified using the security Access process, in applicative software<br>• Implementation of the DID F013 to manage the secondary SW.<br>• Description of the DID life (F012, F188, F013)<br><br>**Details**<br>**Modification of :**<br>RuleReference_vehicleManufacturerSparePartNumber_R_200_b<br>RuleReference_VehicleManufacturerECUHardwareNumber_DAI_200_b<br>RuleReference_vehicleManufacturerSparePartNumber_R_020_b<br>RuleReference_vehicleManufacturerSparePartNumber_R_030_b<br>RuleReference_vehicleManufacturerSparePartNumber_N_030_b<br>RuleReference_vehicleManufacturerSparePartNumber_N_035_b<br>RuleReference_VehicleManufacturerECUHardwareNumber_DAI_040_b<br>RuleReference_VehicleManufacturerECUHardwareNumber_DAI_050_b<br>RuleReference_OperationnalReference_100_b<br>RuleReference_OperationnalReference_110_b<br>RuleReference_OperationnalReference_300_b<br>RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_000_b<br>RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_300_b<br>RuleReference_Fingerprint_000_b<br>RuleReference_Fingerprint_020_b<br>RuleReference_Fingerprint_100_b<br>RuleReference_Fingerprint_200_b<br>RuleReference_BootVersion_001_b<br>RuleReference_BootVersion_010_b<br>RuleReference_BootVersion_015_b<br>RuleReference_BootVersion_020_b<br>RuleReference_BootVersion_025_b<br>RuleReference_BootVersion_030_b<br>RuleReference_BootVersion_300_b<br>RuleReference_CalibrationNumber_140_b<br>RuleReference_CalibrationNumber_300_b<br>RuleReference_ConfigurationFileReferenceLink_120_b<br>RuleReference_Digest_020_b |

| | | |
|---|---|---|
| | | RuleReference_Digest_200_b<br>RuleReference_Fingerprint_110_b<br>RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_015_b<br>RuleReference_SystemNameOrEngineType_050_b<br>RuleReference_VIN_005_b<br>RuleReference_VIN_030_b<br>RuleReference_VIN_040_b<br>RuleReference_VIN_050_b<br>RuleReference_VIN_100_b<br>RuleReference_VIN_300_b<br>RuleReference_ECUSerialNumberDataIdentifier_300_b<br>RuleReference_vehicleManufacturerECUHardwareNumber_000_b<br>RuleReference_ConfigurationFileReferenceLink_100_b<br>RuleReference_Digest_300_b<br>RuleReference_Fingerprint_300_b<br>RuleReference_VDIAG_300_b<br>RuleReference_vehicleManufacturerECUHardwareNumber_110_b<br>RuleReference_Digest_030_b<br>RuleReference_Digest_100_b<br>RuleReference_ConfigurationFileReferenceLink_020_b<br>RuleReference_ConfigurationFileReferenceLink_030_b<br>RuleReference_BootVersion_110_b<br>RuleReference_vehicleManufacturerKitAssemblyPartNumber_110_b<br>RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_110_b<br>RuleReference_BootVersion_110_b<br>RuleReference_CalibrationNumber_110_b<br>RuleReference_CalibrationNumber_120_b<br>RuleReference_systemSupplierIdentifier_100_b<br>RuleReference_SystemSupplierECUsoftwareNumber_110_b<br>RuleReference_SystemSupplierECUSoftwareVersionNumber_110_b<br>RuleReference_VDIAG_100_b<br>RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_100_b<br>RuleReference_SystemNameOrEngineType_100_b<br>RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_300_b<br>**Addition of:**<br>RuleReference_OperationnalReference_115_a<br>RuleReference_OperationnalReference_210_a<br>Chapter Secondary operational reference (F0.13)<br>RuleReference_ConfigurationFileReferenceLink_200_a<br>RuleReference_Fingerprint_012_a<br>RuleReference_Fingerprint_015_a<br>RuleReference_Fingerprint_250_a<br>RuleReference_Fingerprint_260_a<br>RuleReference_Fingerprint_265_a<br>RuleReference_Digest_015_a<br>Chapter IndexSrvData (F0.11)<br>RuleReference_BootVersion_140_a<br>RuleReference_CalibrationNumber_150_a<br>RuleReference_systemSupplierIdentifier_022_a<br>RuleReference_systemSupplierIdentifier_023_a<br>RuleReference_SystemSupplierECUsoftwareNumber_016_a<br>RuleReference_VIN_070_a<br>RuleReference_VIN_060_a<br>RuleReference_VIN_080_a<br>RuleReference_ConfigurationFileReferenceLink_015_a<br>RuleReference_SystemSupplierECUSoftwareVersionNumber_016_a<br>RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_010_b<br>RuleReference_VDIAG_010_a<br>RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_020_a<br>RuleReference_ConfigurationFileReferenceLink_112_a<br>RuleReference_ConfigurationFileReferenceLink_050_a<br>RuleReference_ConfigurationFileReferenceLink_060_a<br>RuleReference_DID_Reserved_RN_001_b<br>**Deletion of:**<br>RuleReference_OperationnalReference_110_b<br>RuleReference_Digest_130_a<br>RuleReference_Digest_040_a (Deleted because already defined in 36-00-029)<br>RuleReference_Digest_045_a<br>RuleReference_Digest_050_a(Deleted because already defined in 36-00-029)<br>RuleReference_ConfigurationFileReferenceLink_110_b |
| January 2015 | --G | **Adding a Glossary**<br><br>**F187SparePartNumber:**<br>modification to have a cross part.<br>**Deletion of** RuleReference_vehicleManufacturerSparePartNumber_R_040_a<br>**Modification of:** |

RuleReference_vehicleManufacturerSparePartNumber_R_100_b
RuleReference_vehicleManufacturerSparePartNumber_R_110_b

**F1A1:SparePartNumber**
modification to have a cross part.
**Deletion of:**
RuleReference_vehicleManufacturerSparePartNumber_N_100_b
**Modification of:**
RuleReference_vehicleManufacturerSparePartNumber_N_110_b
RuleReference_vehicleManufacturerSparePartNumber_N_120_b

**F18E: KitAssembly:**
modification to have a cross part.
**Modification of:**
RuleReference_vehicleManufacturerKitAssemblyPartNumber_100_b

**F191 HW number:**
**Modification of:**
RuleReference_vehicleManufacturerECUHardwareNumber_100_b

**F012 OperationalReference:**
**Deletion of:**
RuleReference_OperationnalReference_210_b
**Modification of:**
RuleReference_OperationnalReference_200_b
**Addition of:**
RuleReference_OperationnalReference_005_a
RuleReference_OperationnalReference_006_a
RuleReference_OperationnalReference_112_a
RuleReference_OperationnalReference_120_a
RuleReference_OperationnalReference_205_a

**F188 RUC:**
This chapter is simplified. The ECU designer has only one choice if the order part is a REFB.
**Deletion of:**
RuleReference_ConfigurationFileReferenceLink_015_a (no needs)
RuleReference_ConfigurationFileReferenceLink_050_a
RuleReference_ConfigurationFileReferenceLink_060_a
RuleReference_ConfigurationFileReferenceLink_112_a
RuleReference_ConfigurationFileReferenceLink_125_b
RuleReference_ConfigurationFileReferenceLink_130_b
RuleReference_ConfigurationFileReferenceLink_200_a
RuleReference_ConfigurationFileReferenceLink_210_b
**Modification of**
RuleReference_ConfigurationFileReferenceLink_030_c
RuleReference_ConfigurationFileReferenceLink_132_b
RuleReference_ConfigurationFileReferenceLink_140_c
RuleReference_ConfigurationFileReferenceLink_150_c
RuleReference_ConfigurationFileReferenceLink_220_b
**Adding of**
RuleReference_ConfigurationFileReferenceLink_005_a
RuleReference_ConfigurationFileReferenceLink_006_a
RuleReference_ConfigurationFileReferenceLink_225_a
RuleReference_ConfigurationFileReferenceLink_226_a

**F1.A2ConfigurationDataReferenceAfterConfigurationProcess:**
Addition of the ConfigurationDataReferenceAfterConfigurationProcess F1.A2: This DID store the configuration file reference.

**DIGEST**
**Modification of:**
RuleReference_Digest_000_b
RuleReference_Digest_110_b
**Addition of:**
RuleReference_Digest_035_a
RuleReference_Digest_040_a

**F180 BootVersion:**
**Modification of**
RuleReference_BootVersion_001_c

**F182 CalibrationNumber:**
Addition remark about the reading of this information in bootloader.

**F18A systemSupplierIdentifier :**
**Modification of**

| | | |
|---|---|---|
| | | RuleReference_systemSupplierIdentifier_022_b<br>RuleReference_systemSupplierIdentifier_023_b<br><br>**F194 SystemSupplierECUsoftwareNumber :**<br>**Deletion of**<br>RuleReference_SystemSupplierECUsoftwareNumber_016_a (no needs)<br><br>**F195SystemSupplierECUSoftwareVersionNumber :**<br>**Deletion of**<br>RuleReference_SystemSupplierECUSoftwareVersionNumber_016_a<br>(no needs)<br><br>**F18C ECUSerialNumberDataIdentifier:**<br>**Modification of:**<br>RuleReference_ECUSerialNumberDataIdentifier_010_b<br>**Addition of:**<br>RuleReference_ECUSerialNumberDataIdentifier_012_a<br>**Deletion of:**<br>RuleReference_VDIAG_015_a (no needs)<br><br>**F196 exhaustRegulationOrTypeApprovalNumberDataIdentifier :**<br>**Modification of:**<br>RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_015_c<br>**Deletion of:**<br>RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_020_a (No needs)<br><br>**F197 SystemNameOrEngineType:**<br>**Deletion of:**<br>RuleReference_SystemNameOrEngineType_020_a (No needs)<br><br>**F190 VIN:**<br>**Deletion of:**<br>RuleReference_VIN_010_a (No Needs)<br>**Modification of:**<br>RuleReference_VIN_060_b<br><br>**F0.D0**: **Mileage of the first occurrence failure**<br>**New DID**<br><br>**F0.D1: DTC Occurence Counter**<br>**New DID** |

**REFERENCED DOCUMENTS**

| | | |
|---|---|---|
| [1] | ISO 14229-1: 2013 | Unified Diagnostic Services : Specification and Requirements |
| [2] | [RENAULT]<br>36-00-027<br>36-00-028<br>36-00-033<br>36-00-029<br>36-00-031<br>36-00-032<br><br>[NISSAN]<br>25953NDS29<br>25953NDS30<br>25953NDS31<br>25953NDS32<br>25953NDS33<br>25953NDS34 | Standard ECU programming Part 1-General description<br>Standard ECU programming Part 2-Ecu-tool programming interfaces description<br>Standard ECU programming Part 3-Boot_Application interface description<br>Standard ECU programming Part 4-Boot-Loader mechanisms<br>Standard ECU programming Part 5-Conformance test<br>Standard UCE programming-PDX file requirements |
| [3] | [RENAULT]<br>36-00-011<br>[NISSAN]<br>25953NDS27 | Unified Diagnostic Services Implementation |
| [4] | [RENAULT]<br>36-00-030<br>[NISSAN]<br>25953NDS35 | Standard Security Access |

# Contents



# Table

# 1 SCOPE AND OBJECTIVES

## 1.1 Scope

The present document contains a set of rules, that help suppliers to manage references in ECU. These rules apply to each ECU which implements UDS diagnostic services REF[3].

## 1.2 Objectives

The purpose of the rules is to define guidelines in order to:

- Define an implementation for references in ECUs
- Be compliant with ISO identifiers,
- Harmonize the various references used in engineer department, the after-sales and plants,

## 1.3 Requirement Terminology

The following terminology is used to define the applicability of each requirement in this document.

- The word **Must** in the text means legislative or regulatory requirements (e.g. Health and Safety).
- The word **Shall** in the text means a mandatory requirement.
- The word **Should** in the text means a recommendation or advice on implementing a requirement. Such recommendations or advice are expected to be followed; unless justified reasons are stated for not doing so.
- The word **Will** in the text means an intention or a provision, in connection with a requirement.
- The word **Can** in the text means a “user defined“ requirement. The OEM designer shall choose how he wants to implement this requirement.

For our use, requirements are noted as follows:

| |
|---|
| [ID]<br>Text requirement<br>See document « reference » *(if needed)* |

**[ID]**: is a unique requirement identifier, which respects the following description:
**[SpecName_XX_YY_nnn_C]**:

- **SpecName**: Short name that gives the specification main field.
- **XX**: Chapter Level 1 short name.
- **YY**: Chapter Level 2 short name
- **nn**: incremental number.
- **C**: requirement version (given by a set of characters)

Text: is the descriptive of the requirement.

« Reference »: is a reference number or indication (if needed).

## 1.4 Requirements

The various references are used to:

- check conformity in plant
- choose the pieces of software or calibration to be downloaded in after-sales.
- choose the configuration file to download in after-sales.
- identify the ECU in order to diagnose it

Note: Ordering replacement parts in after-sales is out of scope. This operation is carried out through an off-board tool, which does not make use of on-board references.
« Reference »: this tag will be used also to know which need is at the origin of this requirement.

## 2 DEFINITIONS

Any object in a vehicle is subject to referencing in the central Renault database, which is called “SIGNE” (or BOM-C, for the Alliance)
There are various reasons for referencing an object, but this is beyond the scope of this document. Generally speaking, an object must be referenced, when its identification must be ensured throughout time and projects. One should keep in mind the different kinds of objects that might be referenced and their relationships.
A PCB is a board with electronic components soldered on it. Though it had never happened, a sole PCB could be referenced in SIGNE.

### 2.1 Glossary

| | |
|---|---|
| UDS | Unified Diagnostic Services |
| DID | Data IDentifier |
| ECU | Electronic Control Unit |
| VIN | Vehicle Identification Number |
| PDX file | Package ODX corresponding to the reprogramming file |
| OEM | Original Equipment Manufacturer. In this document, Renault and Nissan |
| DAI | DAImler |
| RUC | Référence Unique Configuration |
| ALLIANCE | Renault-Nissan Alliance |
| ISO | International Standardization Organisation |
| BOM-C | Alliance IT System which manage reference |
| SIGNE | Renault IT System which manage reference |
| PCB | Printed Circuit Boards |
| EPS | Electronic Power Steering |
| RN | Renault Nissan |
| IT SYSTEM | Information Technology system |
| DTC | Diagnostic Trouble Code |
| ECM | Engine Controller Module |
| OBD | On Board Diagnostic |

### 2.2 ECU Flashable

An ECU is “flashable” when it implements a bootloader software to manage the downloading of applicative software and calibration.

### 2.3 Assembly part

An assembly part corresponds to an electronic part with a mechanical part. For example, the EPS (Electronic Power Steering) consists of two parts, one is the Steering column and the second one is the ECU. The Both part correspond to an assembly part. This part can be ordered by plant.

This reference must be referenced in OEM IT system (ex: in Renault side ➔SIGNE).
This reference will be an “Alliance” reference, defined by Renault or Nissan (depending on who is the main leader of the part). For one part, we will have a unique reference used by Renault and Nissan

| KITassembly ➔ F18E (used in plant to check if it is the right part) | |
|---|---|
| | F187➔ Electronic part (used in after sale to download the right configuration file) |
|  |  |

### 2.4 ORDER PART

An "Order part" is the part ordered by Renault or NISSAN from the supplier and delivered to the plant or after sales. This part may be an assembly. It may be a reprogrammable board, waiting for an application, or just calibration. It cannot be pure software.

This reference must be referenced in OEM IT system (ex: in Renault side ➔SIGNE).
This reference will be an “Alliance” reference, defined by Renault or Nissan (depending on who is the main leader of the part). For one part, we will have a unique reference used by Renault and Nissan

### 2.5 « REFA » reference

This reference corresponds to an electronic part with a **PCB** with or without bootloader. This part is "**not operational**"

This reference must be referenced in OEM IT system (ex: in Renault side ➔SIGNE).
This reference will be an "Alliance" reference, defined by Renault or Nissan (depending on who is the main leader of the part). For one part, we will have a unique reference used by Renault and Nissan

### 2.6 « REFB » reference

This reference corresponds to:

- For an Electronic part
  - a "REFA" + an applicative software. This part is "**not operational**".
- For an Assembly part
  - A "REFA" + an applicative software + a mechanic part. This assembly part is "**not operationa**l".

This reference must be referenced in OEM IT system (ex: in Renault side ➔SIGNE).
This reference will be an "Alliance" reference, defined by Renault or Nissan (depending on who is the main leader of the part). For one part, we will have a unique reference used by Renault and Nissan

### 2.7 « REFC » reference

This reference can corresponds to a

- "REFA" part with a boot + an applicative software + a calibration.
- "REFA" part with a boot + an applicative software and NO calibration, because not need
- "REFA" part without boot + an applicative software + a calibration
- "REFA" part without boot + an applicative software and NO calibration, because not need

In all of these cases, this part is **"operational**".

This reference must be referenced in OEM IT system (ex: in Renault side ➔SIGNE).
This reference will be an "Alliance" reference, defined by Renault or Nissan (depending on who is the main leader of the part). For one part, we will have a unique reference used by Renault and Nissan

### 2.8 Pdx file reference

This reference corresponds to pdx file reference.
The pdx file contains the binary of the SW and the binary of the calibration (If the ECU has a calibration)
This reference allows, for a "REFB" part, to switch from "**not operation**" to" **operational**".

This reference must be referenced in OEM IT system (ex: in Renault side ➔SIGNE).
This reference will be an "Alliance" reference, defined by Renault or Nissan (depending on who is the main leader of the part). For one pdx file, we will have a unique reference used by Renault and Nissan.

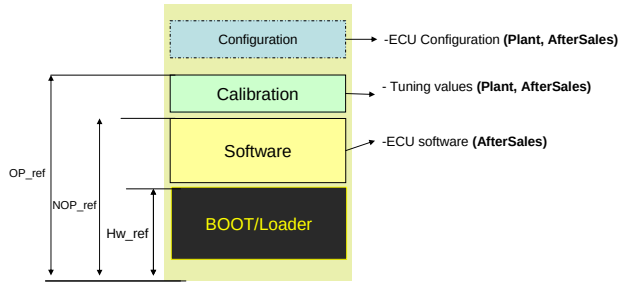
### 2.9 Operational and non operational reference

The term “operational reference” is specific to UDS Renault/Nissan implementation. Operational reference is reference code which represents the ECU finished according programming process.
Non operational reference refers to a system which needs some piece of software to be downloaded in it in order to run properly. A part in state “REFB” is a particular case of a non operational part.
Both are references to physical parts, not pure software.

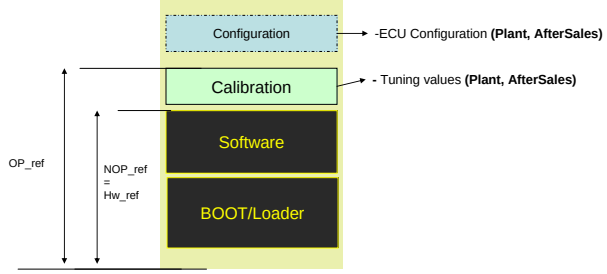
Examples:

NOP_ref means Not Operational reference ECU
OP_ref means Operational reference ECU
HW_ref means Hardware reference ECU

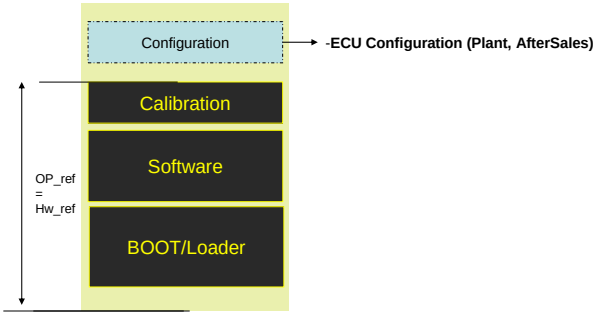
Case 1: OEM can download software & calibration




Case 2 : OEM can download calibration




Case 3 : OEM can’t be able to programming ECU



## 3 DID DEFINITION SYNTHESIS

### 3.1 Generic DID used by OEM

The following table resumes the supported DID which are supported by the applicative software and/or the Bootloarder software according to this document:

☑ : Mandatory
☑ * : Mandatory for OBD ECU
☒ : Forbidden
☐ : User defined and not verified by diag test tool
●: User defined and verified by diag test tool

| DID | Dat; a | Applicatif software | | | | Bootloader software | | | | Snapshot information |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Session | | | | Session | | | | |
| | | Default | | Extended | | Default | | Reprog | | |
| | | Read ($22) | Write ($2E) | Read ($22) | Write ($2E) | Read ($22) | Write ($2E) | Read ($22) | Write $2E) | |
| $F0.D1 | DTC occurrence Counter | ☐ | ☒ | ☐ | ☒ | ☐ | ☒ | ☐ | ☒ | ☑ |
| $F0.10 | Fingerprint | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F0.11 | IndexSrvData | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F0.12 | Operationnal reference | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F0.13 | Secondary operational reference | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F0.D0 | Mileage | ☐ | ☒ | ☐ | ☒ | ☐ | ☒ | ☐ | ☒ | ☑ |
| $F1.11 | VehicleManufacturerECUHardwareNumber_DAI | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F1.21 | VehicleManufacturerECUSoftwareNumber_DAI | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F1.80 | BootVersion | ☒ | ☒ | ☒ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F1.82 | CalibrationNumber | ● | ☒ | ● | ☒ | ☒ | ☒ | ☒ | ☒ | ☒ |
| $F1.87 | VehicleManufacturerSparePartNumber_R | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F1.88 | ConfigurationFileReferenceLink | ☑ | ☒ | ☑ | ☒ | ☐ | ☒ | ☐ | ☒ | ☒ |
| $F1.8A | SystemSupplierIdentifier | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F1.8C | ECUSerialNumberDataIdentifier | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F1.8E | VehicleManufacturerKitAssemblyPartNumber | ● | ☒ | ● | ☒ | ● | ☒ | ● | ☒ | ☒ |
| $F1.90 | VIN | ☑ | ☒ | ☑ | ☑ | ☐ | ☒ | ☐ | ☒ | ☒ |
| $F1.91 | VehicleManufacturerECUHardwareNumber | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F1.94 | SystemSupplierECUsoftwareNumber | ☑ | ☒ | ☑ | ☒ | ☐ | ☒ | ☐ | ☒ | ☒ |
| $F1.95 | SystemSupplierECUSoftwareVersionNumber | ☑ | ☒ | ☑ | ☒ | ☐ | ☒ | ☐ | ☒ | ☒ |
| $F1.96 | exhaustRegulationOrTypeApprovalNumberDataIdentifier | ☑ * | ☒ | ☑ * | ☒ | ☐ | ☒ | ☐ | ☒ | ☒ |
| $F1.97 | SystemNameOrEngineType | ☑ * | ☒ | ☑ * | ☒ | ☐ | ☒ | ☐ | ☒ | ☒ |
| $F1.A0 | VDIAG | ☑ | ☒ | ☑ | ☒ | ☒ | ☒ | ☒ | ☒ | ☒ |
| $F1.A1 | VehicleManufacturerSparePartNumber_N | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☑ | ☒ | ☒ |
| $F1.A2 | ConfigurationDataReferenceAfterConfigurationProcess | ☑ | ☒ | ☑ | ☑ | ☐ | ☒ | ☐ | ☒ | ☒ |
| $FD00-FEFF | Digest | ☒ | ☒ | ☒ | ☒ | ☑ | ☒ | ☑ | ☑ | ☒ |

## 4 DID RESERVED FOR DAIMLER

**RuleReference_DID_Reserved_DAI_001_b**

The DID **F131**, **F150**, **F151**, **F153**, **F154**, **F155** are already used by ECM ECU designed for Daimler and it shall **not** use by the ECU designed ONLY for ALLIANCE.

## 5 DID RESERVED FOR RENAULT/NISSAN

**RuleReference_DID_Reserved_RN_001_b**

The ranges F0.10 - F0.FF and F1.A0 - F1.EF and DA.00 – DA.AF are reserved for the diagnostic department needs. Do not use without agreement of Department Diagnosis.

## 6 DID RESERVED FOR MULTIPLEXED NETWORK DIAGNOSTIC

**RuleReference_DID_Reserved_NETWORK_001_a**

The DID defined in the following table, shall be reserved for the multiplexed network diagnostic.

| Name | DataIdentifier |
|---|---|
| CANV_DiagmuxState | 0xEF82 |
| CANMM85_DiagmuxState | 0xEF85 |
| CANMM86_DiagmuxState | 0xEF86 |
| CANLSFT_DiagmuxState | 0xEF87 |
| CANEL_DiagmuxState | 0xEF88 |
| LINRLSW_DiagmuxState | 0xEF89 |
| LINDOOR_DiagmuxState | 0xEF8A |
| LINEVC_DiagmuxState | 0xEF8B |
| LINBATT_DiagmuxState | 0xEF8C |
| LINALT_DiagmuxState | 0xEF8D |
| CANADAS_DiagmuxState | 0xEF8E |
| multi CAN DiagmuxState (Nissan company) | 0xEF8F |
| CAN**xx**_DiagmuxState | 0xEF90 to 0xEF95 (reserved) |
| LIN diag Frame | 0xEF96 (reserved by document |
| LIN**xx**_DiagmuxState | 0xEF97 to 0xEF9F (reserved) |
| CANV_EcuList | 0xEFB1 |
| CANV_DiagnosableEcuList | 0xEFB2 |
| CANMM_EcuList | 0xEFB3 |
| CANMM_DiagnosableEcuList | 0xEFB4 |
| CANLSFT_EcuList | 0xEFB5 |
| CANLSFT_DiagnosableEcuList | 0xEFB6 |
| CANEL_EcuList | 0xEFB7 |
| CANEL_DiagnosableEcuList | 0xEFB8 |
| CANADAS_EcuList | 0xEFB9 |
| CANADAS_DiagnosableEcuList | 0xEFBA |
| CAN**xx**_EcuList | 0xEFBB,0xEFBD,0xEFBF (reserved) |
| CAN**xx**_DiagnosableEcuList | 0xEFBC,0xEFBE,0xEFC0 (reserved) |

*Remark: For length, format and applicability of multiplexed network diagnostic data, see "Communication Requirement Specification" file from manufacturer ECU designers.*

# 7 ECU REFERENCES MANAGEMENT

## 7.1 VehicleManufacturerSparePartNumber_R (F1.87)

**RuleReference_vehicleManufacturerSparePartNumber_R_000_b**
The **vehicleManufacturerSparePartNumber_R** (**F1.87**) shall be implemented by all ECU designed for ALLIANCE.

### 7.1.1 Read/Write operations

**RuleReference_vehicleManufacturerSparePartNumber_R_001_a**
For a flashable ECU, in default session of bootloader software, the ECU shall answer positively to **READ** the **vehicleManufacturerSparePartNumber_R** ($22.**F1.87**), using the dataRecord parameter as defined in [7.1.3].

**RuleReference_vehicleManufacturerSparePartNumber_R_005_a**
For a flashable ECU, in programming session of bootloader software, the ECU shall answer positively to **READ** the **vehicleManufacturerSparePartNumber_R** ($22.**F1.87**), using the dataRecord parameter as defined in [7.1.3].

**RuleReference_vehicleManufacturerSparePartNumber_R_010_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **vehicleManufacturerSparePartNumber_R** ($22.**F1.87**), using the dataRecord parameter as defined in [7.1.3].

**RuleReference_vehicleManufacturerSparePartNumber_R_015_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **vehicleManufacturerSparePartNumber_R** ($22.**F1.87**), using the dataRecord parameter as defined in [7.1.3].

**RuleReference_vehicleManufacturerSparePartNumber_R_020_b**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **vehicleManufacturerSparePartNumber_R** ($2E.**F1.87**), with Error code 0x31 (7F.2E.31).

**RuleReference_vehicleManufacturerSparePartNumber_R_030_b**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **vehicleManufacturerSparePartNumber** ($2E.**F1.87**), with Error code 0x31 (7F.2E.31).

### 7.1.2 Initialization

**RuleReference_vehicleManufacturerSparePartNumber_R_100_b**
If the reference part is **NOT** defined as an “assembly part” (see chapter 2.3), the **vehicleManufacturerSparePartNumber_R** DataIdentifier (**F1.87**) shall correspond to the OEM order part number reference, as defined in [2.4], and is already equals to the reference defined in the **vehicleManufacturerSparePartNumber_N** DataIdentifier (**F1.A1**).

**RuleReference_vehicleManufacturerSparePartNumber_R_110_b**
If the reference part is defined as an “assembly part” (see chapter 2.3), the **vehicleManufacturerSparePartNumber_R** DataIdentifier (**F1.87**) shall correspond to the OEM reference of the electronic part given by the OEM ECU designer, as defined in [2.4], and is already equals to the reference defined in the **vehicleManufacturerSparePartNumber_N** DataIdentifier (**F1.A1**).

### 7.1.3 OEM programming process behavior

**RuleReference_vehicleManufacturerSparePartNumber_R_200_c**
Any modification of **vehicleManufacturerSparePartNumber_R** (**F1.87)** shall be forbidden by OEM tools.
Flashing new software or calibration, by OEM tool, does not any effect on this information.
Only the supplier process shall be authorized to modify it.

*Remark: The following table explains the life behavior of the vehicleManufacturerSparePartNumber_R (F1.87)*

| Part Ordered | Action | Value after action | Modification after action |
|---|---|---|---|
| ECU in REFA (without SW) | Delivered by supplier | REFA (see 2.5)<br>OR<br>Reference of the electronic part(in case of the F18E is implemented) | NO |
| ECU in REF A (without SW) | Download SW | Initial contents | NO |
| ECU in REFB (with SW) | Delivered by supplier | REFB (see 2.6)<br>OR<br>reference of the electronic part(in case of the F18E is implemented) | NO |
| ECU in REFB (with SW) | Download calibration | Initial contents | NO |
| ECU in REFC (with SW and calibration) | Delivered by supplier | REFC (see 2.7)<br>OR<br>reference of the electronic part (in case of the F18E is implemented) | NO |
| ECU in REFC (with SW and calibration) | Download only a new SW | Initial contents | NO |
| ECU in REFC (with SW and calibration) | Download only a new calibration | Initial contents | NO |
| ECU in REFC (with SW and calibration) | Delete SW | Initial contents | NO |

**Table 1:** vehicleManufacturerSparePartNumber_R (F1.87) life behavior

### 7.1.4 dataRecord definition

**RuleReference_vehicleManufacturerSparePartNumber_R_300_a**
The ECU shall use the **vehicleManufacturerSparePartNumber_R** (**F1.87**) dataRecord parameter, as defined in the following table.

| Size (bytes) | Format | Coding | Forbidden Character | Data defined by |
|---|---|---|---|---|
| 10 fix | ASCII | Example:<br>$34$37$39$35$32$30$30$32$33$52 = “479520023R” | - | OEM |

### 7.2 VehicleManufacturerSparePartNumber_N (F1.A1)

**RuleReference_vehicleManufacturerSparePartNumber_N_000_b**
The **vehicleManufacturerSparePartNumber_N** (**F1.A1**) shall be implemented by all ECU designed for ALLIANCE.

#### 7.2.1 Read/Write operations

**RuleReference_vehicleManufacturerSparePartNumber_N_010_a**
For a flashable ECU, in default session of bootloader software, the ECU shall answer positively to **READ** the **vehicleManufacturerSparePartNumber_N** ($22.**F1.A1**), using the dataRecord parameter as defined in [7.2.3].

**RuleReference_vehicleManufacturerSparePartNumber_N_015_a**
For a flashable ECU, in programming session of bootloader software, the ECU shall answer positively to **READ** the **vehicleManufacturerSparePartNumber_N** ($22.**F1.A1**), using the dataRecord parameter as defined in [7.2.3].

**RuleReference_vehicleManufacturerSparePartNumber_N_020_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **vehicleManufacturerSparePartNumber_N** ($22.**F1.A1**), using the dataRecord parameter as defined in [7.2.3].

**RuleReference_vehicleManufacturerSparePartNumber_N_025_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **vehicleManufacturerSparePartNumber_N** ($22.**F1.A1**), using the dataRecord parameter as defined in [7.2.3].

**RuleReference_vehicleManufacturerSparePartNumber_N_030_b**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **vehicleManufacturerSparePartNumber_N** ($2E.**F1.A1**), with Error code 0x31 (7F.2E.31).

**RuleReference_vehicleManufacturerSparePartNumber_N_035_b**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **vehicleManufacturerSparePartNumber_N** ($2E.**F1.A1**), with Error code 0x31 (7F.2E.31).

#### 7.2.2 Initialization

**RuleReference_vehicleManufacturerSparePartNumber_N_110_b**
If the reference part is **NOT** defined as an “assembly part” (see chapter 2.3), the **vehicleManufacturerSparePartNumber_N** DataIdentifier (**F1.A1**) shall correspond to the OEM order part number reference, as defined in [2.4], and is already equals to the reference defined in the **vehicleManufacturerSparePartNumber_R** DataIdentifier (**F1.87**).

**RuleReference_vehicleManufacturerSparePartNumber_N_120_b**
If the reference part is defined as an “assembly part” (see chapter 2.3), the **vehicleManufacturerSparePartNumber_N** DataIdentifier (**F1.A1**) shall correspond to the OEM reference of the electronic part given by the OEM ECU designer, as defined in [2.4], and is already equals to the reference defined in the **vehicleManufacturerSparePartNumber_R** DataIdentifier (**F1.87**).

### 7.2.3 OEM programming process behavior

**RuleReference_vehicleManufacturerSparePartNumber_N_200_a**
Any modification of **vehicleManufacturerKitAssemblyPartNumber_N (F1.A1)** shall be forbidden by OEM tools. Only the supplier process shall be authorized to modify it.

*Remark:*

- *Only the supplier knows which couple of hardware and software corresponds to a part reference ordered by OEM*
- *Flashing new software or calibration, by OEM tool, does not any effect on this information.*

### 7.2.4 dataRecord definition

**RuleReference_vehicleManufacturerSparePartNumber_N_300_a**
The ECU shall use the **vehicleManufacturerSparePartNumber_N** (**F1.A1**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 10 Fix | ASCII | Example:<br>$34$37$39$35$32$30$30$32$33$4E= 479520555N | - | OEM |

### 7.3 VehicleManufacturerKitAssemblyPartNumber (F1.8E)

Which ECU implement it: If a part includes an electronic part and a mechanical part and if the OEM part designer needs to distinguish the both reference parts because the electronic part can be changed, in after sale, without change the mechanical part.

Remark: If the reference reading by F1.87 DID and F1.A1 DID equals to reference reading by F1.8E DID, so the F1.8E DID is not necessary.

#### 7.3.1 Read/Write operations

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_001_a**
For a flashable ECU, in default session of bootloader software, the ECU shall answer positively to **READ** the **vehicleManufacturerKitAssemblyPartNumber** ($22.**F1.8E**), using the dataRecord parameter as defined in [7.3.3],
**IF** ECU belongs to an assembly part **AND**
the ECU designer needs to distinguish the electronic part reference and the assembly part reference
**ELSE** it shall answer negatively with Error code 0x31 (7F.22.31).

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_005_a**
For a flashable ECU, in programming session of bootloader software, the ECU shall answer positively to **READ** the **vehicleManufacturerKitAssemblyPartNumber** ($22.**F1.8E**), using the dataRecord parameter as defined in [7.3.3],
**IF** ECU belongs to an assembly part **AND**
the ECU designer needs to distinguish the electronic part reference and the assembly part reference
**ELSE** it shall answer negatively with Error code 0x31 (7F.22.31).

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_010_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **vehicleManufacturerKitAssemblyPartNumber** ($22.**F1.8E**), using the dataRecord parameter as defined in [7.3.3],
**IF** ECU belongs to an assembly part **AND**
the ECU designer needs to distinguish the electronic part reference and the assembly part reference
**ELSE** it shall answer negatively with Error code 0x31 (7F.22.31).

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_012_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **vehicleManufacturerKitAssemblyPartNumber** ($22.**F1.8E**), using the dataRecord parameter as defined in [7.3.3],
**IF** ECU belongs to an assembly part **AND**
the ECU designer needs to distinguish the electronic part reference and the assembly part reference
**ELSE** it shall answer negatively with Error code 0x31 (7F.22.31).

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_020_a**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **vehicleManufacturerKitAssemblyPartNumber** ($2E.**F1.8E**), with Error code 0x31 (7F.2E.31).

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_025_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **vehicleManufacturerKitAssemblyPartNumber** ($2E.**F1.8E**), with Error code 0x31 (7F.2E.31).

### 7.3.2 Initialization

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_100_b**
The **vehicleManufacturerKitAssemblyPartNumber** DataIdentifier (**F1.8E**) shall correspond to the OEM order Kit Assembly part number reference, as defined in [2.3].

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_110_b**
When delivered to OEM, the **vehicleManufacturerKitAssemblyPartNumber** (**F1.8E**) DID shall be initialized according to [7.3.4].

### 7.3.3 OEM programming process behavior

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_030_a**
Any modification of **vehicleManufacturerKitAssemblyPartNumber** ***(F1.8E)*** shall be forbidden by OEM tools. Only the supplier process shall be authorized to modify it.

*Remark:*

- *If the ECU part and the mechanical part are assembled in supplier plant, the supplier tool plant writes in EOL, the new kit assembly reference. So the OEM tool must not modify it, because OEM does not have information on whether two parts can be paired together.*
- *Flashing new software or a calibration, by OEM tool, does not any effect on this information.*

*Remark: The following table explains the life behavior of the vehicleManufacturerKitAssemblyPartNumber (F1.8E)*

| *Initial State* | *Action* | *Value after action* | *Modification after action* |
|---|---|---|---|
| ECU in REFA (without SW) | Delivered by supplier | OEM reference of the part ordered | NO |
| ECU in REF A (without SW) | Download SW | Initial contents | NO |
| ECU in REFB (with SW) | Delivered by supplier | OEM reference of the part ordered | NO |
| ECU in REFB (with SW) | Download calibration | Initial contents | NO |
| ECU in REFC (with SW and calibration) | Delivered by supplier | OEM reference of the part ordered | NO |
| ECU in REFC (with SW and calibration) | Download only a new SW | Initial contents | NO |
| ECU in REFC (with SW and calibration) | Download only a new calibration | Initial contents | NO |
| ECU in REFC (with SW and calibration) | Delete SW | Initial contents | NO |

**Table 2:** vehicleManufacturerKitAssemblyPartNumber (F1.8E) life behavior

### 7.3.4 dataRecord definition

**RuleReference_vehicleManufacturerKitAssemblyPartNumber_200_a**
The ECU shall use the **vehicleManufacturerKitAssemblyPartNumber** (**F1.8E)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 10 Fix | ASCII | Example:<br>$34$37$39$35$32$30$30$32$33$52 = 479520023R | - | OEM |

### 7.4 VehicleManufacturerECUHardwareNumber (F1.91)

This DID correspond to the Hwd_Ref defined in chapter [2.9]. It identifies the ECU hardware AND bootloader software.

**RuleReference_vehicleManufacturerECUHardwareNumber_000_c**
The **vehicleManufacturerECUHardwareNumber** (**F1.91)** shall be implemented by all ECU designed for ALLIANCE.

#### 7.4.1 Read/Write operations

**RuleReference_vehicleManufacturerECUHardwareNumber_005_a**
For a flashable ECU, in default session of bootloader software, the ECU shall answer positively to **READ** the **vehicleManufacturerECUHardwareNumber** ($22.**F1.91**), using the dataRecord parameter as defined in [7.4.4].

**RuleReference_vehicleManufacturerECUHardwareNumber_010_a**
For a flashable ECU, in programming session of bootloader software, the ECU shall answer positively to **READ** the **vehicleManufacturerECUHardwareNumber** ($22.**F1.91**), using the dataRecord parameter as defined in [7.4.4].

**RuleReference_vehicleManufacturerECUHardwareNumber_020_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **vehicleManufacturerECUHardwareNumber** ($22.**F1.91**), using the dataRecord parameter as defined in [7.4.4].

**RuleReference_vehicleManufacturerECUHardwareNumber_025_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **vehicleManufacturerECUHardwareNumber** ($22.**F1.91**), using the dataRecord parameter as defined in [7.4.4].

**RuleReference_vehicleManufacturerECUHardwareNumber_030_a**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **vehicleManufacturerECUHardwareNumber** ($2E.**F1.91**), with Error code 0x31 (7F.2E.31).

**RuleReference_vehicleManufacturerECUHardwareNumber_035_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **vehicleManufacturerECUHardwareNumber** ($2E.**F1.91**), with Error code 0x31 (7F.2E.31).

### 7.4.2 Initialization

**RuleReference_vehicleManufacturerECUHardwareNumber_100_b**
A non-flashable ECU can answer to reading **vehicleManufacturerECUHardwareNumber** DataIdentifer ($22.**F1.91**) with the same data as order part number, as defined in [2.4].

**RuleReference_vehicleManufacturerECUHardwareNumber_110_b**
For a flashable ECU, when delivered to OEM, the **vehicleManufacturerECUHardwareNumber** (**F1.91**) DID shall be initialized according to [7.4.4].

### 7.4.3 OEM programming process behavior

**RuleReference_vehicleManufacturerECUHardwareNumber_200_a**
Any modification of **vehicleManufacturerECUHardwareNumber** ***(F1.91)*** shall be forbidden by OEM tools. Only the supplier process shall be authorized to modify it.

*Remark: The following table explains the life behavior of the vehicleManufacturerECUHardwareNumber (F1.91)*

| *Initial State* | *Action* | *Value after action* | *Modification after action* |
|---|---|---|---|
| ECU in REFA (without SW) | Delivered by supplier | REFA (see 2.5) | NO |
| ECU in REF A (without SW) | Download SW | REFA (see 2.5) | NO |
| ECU in REFB (with SW) | Delivered by supplier | REFA (see 2.5) | NO |
| ECU in REFB (with SW) | Download calibration | REFA (see 2.5) | NO |
| ECU in REFC (with SW and calibration) | Delivered by supplier | REFA (see 2.5) | NO |
| ECU in REFC (with SW and calibration) | Download only a new SW | REFA (see 2.5) | NO |
| ECU in REFC (with SW and calibration) | Download only a new calibration | REFA (see 2.5) | NO |
| ECU in REFC (with SW and calibration) | Delete SW | REFA (see 2.5) | NO |
| ECU in REFC (non flashable) | Delivered by supplier | OEM reference of the part ordered | NO |

**Table 3:** vehicleManufacturerECUHardwareNumber (F1.91)life behavior

### 7.4.4 dataRecord definition

**RuleReference_vehicleManufacturerECUHardwareNumber_300_a**
The ECU shall use the **vehicleManufacturerECUHardwareNumber** (**F1.91)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 10 Fix | ASCII | Example:<br>$34$37$39$35$31$30$30$32$33$52 = 478510023R | - | OEM |

### 7.5 VehicleManufacturerECUHardwareNumber_DAI (F1.11)

**RuleReference_VehicleManufacturerECUHardwareNumber_DAI_000_a**
The **VehicleManufacturerECUHardwareNumber_DAI** (**F1.11**) shall be implemented **ONLY** for all ECU designed for Daimler.

#### 7.5.1 Read/Write operations

**RuleReference_VehicleManufacturerECUHardwareNumber_DAI_010_a**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer positively to **READ** the **VehicleManufacturerECUHardwareNumber_DAI** ($22.**F1.11**), using the dataRecord parameter as defined in [7.5.4].

**RuleReference_VehicleManufacturerECUHardwareNumber_DAI_020_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **VehicleManufacturerECUHardwareNumber_DAI** ($22.**F1.11**), using the dataRecord parameter as defined in [7.5.4].

**RuleReference_VehicleManufacturerECUHardwareNumber_DAI_030_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **VehicleManufacturerECUHardwareNumber_DAI** ($22.**F1.11**), using the dataRecord parameter as defined in [7.5.4].

**RuleReference_VehicleManufacturerECUHardwareNumber_DAI_040_b**
For an ECU flashable, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **VehicleManufacturerECUHardwareNumber_DAI** ($2E.**F1.11**), with Error code 0x31 (7F.2E.31).

**RuleReference_VehicleManufacturerECUHardwareNumber_DAI_050_b**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **VehicleManufacturerECUHardwareNumber_DAI** ($2E.**F1.11**), with Error code 0x31 (7F.2E.31).

#### 7.5.2 Initialization

**RuleReference_vehicleManufacturerECUHardwareNumber_DAI_100_a**
When delivered to OEM, the **vehicleManufacturerECUHardwareNumber_DAI** (**F1.11**) DID shall be initialized according to [7.5.4].

#### 7.5.3 OEM programming process behavior

**RuleReference_VehicleManufacturerECUHardwareNumber_DAI_200_b**
Any modification of **vehicleManufacturerECUHardwareNumber_DAI** ***(F1.11)*** shall be forbidden by OEM tools. Only the supplier process shall be authorized to modify it.

#### 7.5.4 dataRecord definition

**RuleReference_VehicleManufacturerECUHardwareNumber_DAI_300_a**
The ECU shall use the **VehicleManufacturerECUHardwareNumber_DAI** (**F1.11**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 10 Fix | ASCII | Example:<br>$38$37$39$39$30$31$30$32$33$34= 879 901 02 34 | - | OEM |

## 7.6 Operationnal reference (F0.12)

The operational reference shall correspond to ECU final state. To simplify, the operational reference equals to the pdx file reference, which will be download in plant or in after sale.

**RuleReference_OperationnalReference_000_b**
The **OperationnalReference** (**F0.12)** shall be implemented by all ECU designed for ALLIANCE.

**RuleReference_OperationnalReference_005_a**
For a flashable ECU, if the ECU has software and calibration, the **OperationnalReference (F012)** value shall be located in a logical block that supports calibration. (see to ref [2])

**RuleReference_OperationnalReference_006_a**
For a flashable ECU, if the ECU has software but no calibration, the **OperationnalReference (F012)** value shall be located in a logical block that supports software. (see to ref [2])

**RuleReference_OperationnalReference_040_a**
The **OperationnalReference** (**F0.12**) shall have a unique relationship with **SystemSupplierECUsoftwareNumber** (**F1.94**) and **SystemSupplierECUSoftwareVersionNumber** (**F1.95**).

### 7.6.1 Read/Write operations

**RuleReference_OperationnalReference_010_a**
For a flashable ECU, in default session of bootloader software, the ECU can answer positively to **READ** the **OperationnalReference** ($22.**F0.12**), using the dataRecord parameter as defined in [7.6.4].

**RuleReference_OperationnalReference_015_a**
For a flashable ECU, in programming session of bootloader software, the ECU can answer positively to the **READ OperationnalReference** ($22.**F0.12**), using the dataRecord parameter as defined in [7.6.4].

**RuleReference_OperationnalReference_020_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **OperationnalReference** ($22.**F0.12**), using the dataRecord parameter as defined in [7.6.4].

**RuleReference_OperationnalReference_025_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **OperationnalReference** ($22.**F0.12**), using the dataRecord parameter as defined in [7.6.4].

**RuleReference_OperationnalReference_030_a**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **OperationnalReference** ($2E.**F0.12**), with Error code 0x31 (7F.2E.31).
**RuleReference_OperationnalReference_035_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **OperationnalReference** ($2E.**F0.12**), with Error code 0x31 (7F.2E.31).

### 7.6.2 Initialization

**RuleReference_OperationnalReference_100_b**
When delivered to OEM, for **NON-**flashable ECU, the **OperationnalReference** (**F0.12**) shall be equal to the order part reference.
*Remark: In this case the **OperationalReference** shall be equal to the **vehicleManufacturerSparePartNumber_R /_N** or the **vehicleManufacturerKitAssemblyPartNumber**. It's depending on the ECU designer.*

**RuleReference_OperationnalReference_112_a**
**WHEN** delivered to OEM, for an ECU flashable **AND** the ECU is delivered without software and without calibration **THEN** the **OperationnalReference** (**F0.12**) DID shall be equal to $30$30$30$30$30$30$30$30$30$30, according to [7.6.4].

**RuleReference_OperationnalReference_115_a**

**WHEN** delivered to OEM, for an ECU flashable **AND** the ECU is delivered without calibration **THEN** the **OperationnalReference** (**F0.12**) DID shall be equal to $30$30$30$30$30$30$30$30$30$30, according to [7.6.4].

**RuleReference_OperationnalReference_120_a**
**WHEN** delivered to OEM, for an ECU flashable **AND** if the ECU is delivered with software and calibration **THEN** the **OperationnalReference** (**F0.12**) DID shall be equal to “pdxfile” reference associated, as defined in [2.7].

### 7.6.3 OEM programming process behavior

**RuleReference_OperationnalReference_200_b**
For an ECU flashable and **NOT** require calibration, when the programming process downloads new software, the **OperationnalReference (F0.12)** DID shall change automatically in accordance with the pdx file reference.
*Remark: the reference of the pdxfile is stored in logical block of the software.*

**RuleReference_OperationnalReference_205_a**
For an ECU flashable and require calibration, when the programming process downloads new calibration, the **OperationnalReference (F0.12)** DID shall change automatically in accordance with the pdx file reference.
*Remark: the reference of the pdxfile is stored in logical block of the calibration.*

*Remark: The following table explains the life behavior of the OperationnalReference (F0.12)*

| Initial State | Action | Value after action | Modification after action |
|---|---|---|---|
| ECU in **REFA** (flashable, without SW) | Delivered by supplier | $30$30$30$30$30$30$30$30$30$30 | NO |
| ECU in **REFA** (flashable, without SW, without calibration) | Delivered by supplier | $30$30$30$30$30$30$30$30$30$30 | NO |
| ECU in **REFA** (flashable, without SW) | Download only a new SW | Reference of the pdx file | YES |
| ECU in **REFA** (flashable, without SW, without calibration) | Download only a new SW | $30$30$30$30$30$30$30$30$30$30 | NO |
| ECU in **REFB** (flashable/SW , without calibration) | Download only the calibration | Reference of the pdx file | YES |
| ECU in **REFC** (flashable/SW/Calibration) | Download only a new calibration | Reference of the new pdx file | YES |
| ECU in **REFC** (flashable/SW)<br>NO calibration, because not need | Download a new SW | Reference of the new pdx file | YES |
| ECU in **REFC** (not flashable/SW/calibration) | Delivered by supplier | Reference of the order part | NO |
| ECU in **REFC** (not flashable/SW)<br>NO calibration, because not need | Delivered by supplier | Reference of the order part | NO |

**Table 4**: OperationnalReference (F0.12) life behavior

### 7.6.4 dataRecord definition

**RuleReference_OperationnalReference_300_b**
The ECU shall use the **OperationnalReference** (**F0.12)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 10 Fix | ASCII | For a non flashable ECU<br>Example: $34$37$36$36$30$30$30$32$33$52 = 476600023R (order part reference)<br><br>For a flashable ECU, without calibration and before programming:<br>$30$30$30$30$30$30$30$30$30$30 = “0000000000”<br><br>For an flashable ECU, with a calibration and after erasing software and calibration:<br>$30$30$30$30$30$30$30$30$30$30 = “0000000000”<br><br>For a flashable ECU, after programming:<br>Example: $34$37$38$35$35$30$30$35$332$52 = “478550052R” ➔ref of the pdx file | - | OEM |

### 7.7 Secondary operational reference (F0.13)

The secondary operational reference shall correspond to second ECU software. To simplify, the secondary operational reference equals to the secondary software file reference delivered by supplier. This file, of the secondary software, will be downloading only in after sale. This file will be a “zip“ file.

**RuleReference_SecondOperRef_000_a**
The **SecondOperRef** (**F0.13)** shall be implemented by all ECU designed for Renault and all ECU designed for Nissan which have two independent software inside the same box which could be updated separately.

#### 7.7.1 Read/Write operations

*Remark: The OEM tool doesn’t read the **SecondOperRef** DID in bootloader software.*
*Whatever the supplier **SecondOperRef** implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [7.7.4].*

**RuleReference_SecondOperRef_010_a**
**IF** an ECU is **not able** to start application mode without its secondary software being operational,
**THEN** it shall provide **SecondOperRef** ($22.**F0.13**) **READING** ability in programming and default sessions of bootloader using the dataRecord parameter as defined in [7.7.4].

**RuleReference_SecondOperRef_020_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **SecondOperRef** ($22.**F0.13**), using the dataRecord parameter as defined in [7.7.4].

**RuleReference_SecondOperRef_025_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **SecondOperRef** ($22.**F0.13**), using the dataRecord parameter as defined in [7.7.4].

**RuleReference_SecondOperRef_030_a**
In all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **SecondOperRef** ($2E.**F0.13**), with Error code 0x31 (7F.2E.31).

**RuleReference_SecondOperRef_035_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **SecondOperRef** ($2E.**F0.13**), with Error code 0x31 (7F.2E.31).

*Remark: This reference corresponds to the reference of the secondary SW file.*

#### 7.7.2 Initialization

**RuleReference_SecondOperRef_100_a**
The **SecondOperRef** (**F0.13**) shall be equal to the reference of the secondary software.

**RuleReference_SecondOperRef_110_a**
IF an ECU has no secondary software yet downloaded, the **SecondOperRef** (**F0.13**) shall be equal to $30$30$30$30$30$30$30$30$30$30.

### 7.7.3 OEM programming process behavior

**RuleReference_SecondOperRef_200_a**
When the programming process downloads a new secondary software, the **SecondOperRef (F0.13)** DID shall change in accordance with the new file reference of the secondary SW.

**RuleReference_SecondOperRef_210_a**
In case of interrupted download of the secondary software erasing, the **SecondOperRef (F0.13)** DID shall be equal to $30$30$30$30$30$30$30$30$30$30.

*Remark: The following table explains the life behavior of the SecondOperRef (F0.13)*

| *Initial State* | *Action* | *Value after action* | *Modification after action* |
|---|---|---|---|
| ECU in REFA (without main SW and without secondary SW) | No action | $30$30$30$30$30$30$30$30$30$30 | NO |
| ECU in REFA (without main SW and without secondary SW) | Download secondary SW | Reference of the secondary SW | YES |
| ECU in REFA (without main SW and without secondary SW) | Download main SW | $30$30$30$30$30$30$30$30$30$30 | NO |
| ECU in REFA (without main SW and with secondary SW) | Download a **new** secondary SW | Reference of the **NEW** secondary SW | YES |
| ECU in REFA (with main SW and with secondary SW) | Download main **new** SW | Reference of the secondary SW | NO |
| ECU in REFA (with main SW and with secondary SW) | Download a new secondary SW | Reference of the **NEW** secondary SW | YES |
| ECU in REFC (with main SW and with secondary SW) | Interruption of the programming of the secondary SW | $30$30$30$30$30$30$30$30$30$30 | YES |

**Table 5**: SecondOperRef (F0.13) life behavior

### 7.7.4 dataRecord definition

**RuleReference_SecondOperRef_300_a**
The ECU shall use the **SecondOperRef** (**F0.13)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 10 Fix | ASCII | An ECU with no secondary software initially downloaded<br>Before programming: $30$30$30$30$30$30$30$30$30$30<br><br>An ECU with a secondary software downloaded<br>Example: $34$37$38$35$35$30$30$32$35$52 = 478550053R | - | OEM |

### 7.8 VehicleManufacturerECUSoftwareNumber_DAI (F1.21)

This reference corresponds to the reference to the pdx file, which will be downloaded in plant or in after sale.

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_000_b**
The **VehicleManufacturerECUSoftwareNumber_DAI** (**F1.21)** shall be implemented ONLY by all ECU designed for Daimler.

#### 7.8.1 Read/Write operations

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_010_b**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer positively to **READ** the **VehicleManufacturerECUSoftwareNumber_DAI** ($22.**F1.21**), using the dataRecord parameter as defined in [7.8.4].

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_020_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **VehicleManufacturerECUSoftwareNumber_DAI** ($22.**F1.21**), using the dataRecord parameter as defined in [7.8.4].

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_025_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **VehicleManufacturerECUSoftwareNumber_DAI** ($22.**F1.21**), using the dataRecord parameter as defined in [7.8.4].

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_030_a**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** **VehicleManufacturerECUSoftwareNumber_DAI** ($2E.**F1.21**), with Error code 0x31 (7F.2E.31).

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_035_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** **VehicleManufacturerECUSoftwareNumber_DAI** ($2E.**F1.21**), with Error code 0x31 (7F.2E.31).

#### 7.8.2 Initialization

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_110_b**
When delivered to OEM, the **VehicleManufacturerECUSoftwareNumber_DAI** (**F1.21**) DID shall be initialized according to [7.8.4].

#### 7.8.3 OEM programming process behavior

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_200_a**
When the programming process downloads new software or calibration, the **VehicleManufacturerECUSoftwareNumber_DAI (F1.21)** DID shall change in accordance with the pdx file reference.

### 7.8.4 dataRecord definition

**RuleReference_VehicleManufacturerECUSoftwareNumber_DAI_300_b**
The ECU shall use the **VehicleManufacturerECUSoftwareNumber_DAI** (**F1.21)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 20 Fix | ASCII | For a flashable ECU<br><br>ECU was never programmed (doesn't contain valid software):<br>$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF<br><br>After programming (incl. supplier process):<br>Example:<br>$38$39$39$39$30$32$30$32$33$34$38$39$39$39$30$32$30$32$33$35 = 89990202348999020235<br><br>If ECU reprogrammable only in development (not documented for plant or after sales):<br>Example:<br>$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30 = 000 000 00 00 000 000 00 00 | - | OEM |

### 7.9 ConfigurationFileReferenceLink-RUC (F1.88)

Needs: This DID is used to download the right configuration in part.

**RuleReference_ConfigurationFileReferenceLink_000_b**
For a configurable ECU, the **ConfigurationFileReferenceLink** (**F1.88)** shall be implemented by all ECU designed for ALLIANCE.

**RuleReference_ConfigurationFileReferenceLink_005_a**
For a flashable ECU, if the ECU has software and calibration, the **ConfigurationFileReferenceLink (F188)** value shall be located in a logical block that supports calibration. (see to ref [2])

**RuleReference_ConfigurationFileReferenceLink_006_a**
For a flashable ECU, if the ECU has software but no calibration, the **ConfigurationFileReferenceLink (F188)** value shall be located in a logical block that supports software. (see to ref [2])

#### 7.9.1 Read/Write operations

**RuleReference_ConfigurationFileReferenceLink_010_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **ConfigurationFileReferenceLink** ($22.**F1.88**), using the dataRecord parameter as defined in [7.9.4].

**RuleReference_ConfigurationFileReferenceLink_020_b**
In extended session of applicative software, the ECU shall answer positively to **READ** the **ConfigurationFileReferenceLink** ($22.**F1.88**), using the dataRecord parameter as defined in [7.9.4].

*Remark: The OEM tool doesn't read the DID in bootloader software.*
*Whatever the supplier **onf**implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [7.9.4].*

**RuleReference_ConfigurationFileReferenceLink_025_a**
In all sessions of bootloader software, the ECU shall answer negatively to **WRITE** **ConfigurationFileReferenceLink** ($2E.**F1.88**), with Error code 0x31 (7F.2E.31).

**RuleReference_ConfigurationFileReferenceLink_030_c**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **ConfigurationFileReferenceLink** ($2E.**F1.88**), with Error code 0x31 (7F.2E.31).

#### 7.9.2 Initialization

**RuleReference_ConfigurationFileReferenceLink_100_b**
If the ECU is non-configurable, the **ConfigurationFileReferenceLink (F1.88)** shall contain the order part reference.

**RuleReference_ConfigurationFileReferenceLink_132_b**
For an ECU, configurable and flashable, with the **F1.88** which depends on the (**REFB**),
*When delivered to OEM, the **ConfigurationFileReferenceLink (F1.88)*** shall contain the (**REFB**).

**RuleReference_ConfigurationFileReferenceLink_140_c**
For an ECU, configurable and flashable, with the **F1.88** which depends on the (**REFC**),
*When delivered to OEM, the **ConfigurationFileReferenceLink (F1.88)*** shall contain the (**pdx file reference**).

**RuleReference_ConfigurationFileReferenceLink_150_c**
For an ECU, configurable and **NOT** flashable, with the F1.88 which depends on the (**REFC**),
*When delivered to OEM, the **ConfigurationFileReferenceLink (F1.88)*** shall contain the (**REFC**).

#### 7.9.3 Reprogramming process Behavior

**RuleReference_ConfigurationFileReferenceLink_220_b**
For an ECU, configurable and flashable, with the **F1.88** which depends on the (**REFB**),

**WHEN** the programming process download new software,
**THEN** the binary of the software will be downloaded.
The binary of the calibration will be downloaded also **BECAUSE** calibration binary will contain new (**REFB)** value.
The **ConfigurationFileReferenceLink** DID shall be equal to the new (**REFB**) depending on this new software and calibration.

**RuleReference_ConfigurationFileReferenceLink_225_a**
For an ECU, configurable and flashable, with the **F1.88** which depends on the (**REFC**) and the software **needs** a calibration to run,
**WHEN** the programming process download new software,
**THEN** the binary of the software is downloaded.
The binary of the calibration will be downloaded also, **BECAUSE** calibration binary will contain new (**Pdx file reference)** value.
The **ConfigurationFileReferenceLink** DID shall be equal to **pdx file reference** depending on this new software and calibration.

**RuleReference_ConfigurationFileReferenceLink_226_a**
For an ECU with the F1.88 depends on the (**REFC**) and the software **doesn't need** calibration to run,
**WHEN** the programming process download new software,
**THEN** the binary of the software will be downloaded, software binary will contain new (**Pdx file reference)** value.
The **ConfigurationFileReferenceLink** DID shall be equal to the **pdx file reference** depending on this new software.

*Remark: The following table explains the life behavior of the ConfigurationFileReferenceLink (F1.88)*

| Use case | Part Ref | Delivered part | F188 value when delivered by supplier | F188 value, after download calibration by OEM processing in plant | F188 value, after download only the SW by OEM processing in after sale | F188 value, after download SW and Calibration by OEM processing in after sale |
|---|---|---|---|---|---|---|
| 1<br>flashable + configurable | REFB | HW + SW | (REFB) | The F188 value, stored in the calibration binary.<br>In this case this value will be the same. | No effect, but after downloading the SW, the calibration will be download automatically and after that the F188 will be modified (see 7.10.3) | F188 is modified automatically when the binary of the SW AND the binary of the calibration are download F188=new REFB (see 7.10.3) |
| 2<br>flashable + configurable | REFC | HW+SW+Calibration | pdx file reference | N/A | No effect, but after downloading the SW, the calibration will be download automatically. (see 7.10.3) | F188 is modified automatically when the binary of the SW AND the binary of the calibration are download F188=Pdx file reference (see 7.10.3) |
| 3<br>flashable + configurable | REFC | HW+SW without calibration because not need | pdx file reference | N/A | F188 is modified automatically when the binary of the SW is download<br>F188=new pdx file reference (see 7.10.3) | N/A |
| 4<br>Not-flashable + configurable | REFC | HW+SW+Calibration | (REFC) | N/A | N/A | N/A |

**Table 6**: ConfigurationFileReferenceLink (F1.88) life behavior

### 7.9.4 dataRecord definition

**RuleReference_ConfigurationFileReferenceLink_300_a**
The ECU shall use the **ConfigurationFileReferenceLink** (**F1.88**) dataRecord parameter, as defined in the following table.
Origin: correspond to Rule G.2 of 36-02-014--C

| Size (bytes) | Format | Coding | Forbidden Character | Data defined by |
|---|---|---|---|---|
| 10 Fix | ASCII | Example:<br>$34$37$39$35$32$30$30$32$33$52 = 479520023R | - | OEM |

### 7.10 ConfigurationDataReferenceAfterConfigurationProcess (F1.A2)

This DID is used to select the right configuration data by the OEM tool. The configuration reference is written by the OEM tool in order to refer this DID after the configuration process.

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_000_a**
For a configurable ECU, the **ConfigurationDataReferenceAfterConfigurationProcess** (**F1.A2)** shall be implemented by all ECU designed for ALLIANCE.

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_005_a**
For a flashable ECU, if the ECU has software and calibration, the **ConfigurationDataReferenceAfterConfigurationProcess** (**F1.A2)** value shall be located in a logical block that supports calibration. (see to ref [2])

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_006_a**
For a flashable ECU, if the ECU has software but no calibration, the **ConfigurationDataReferenceAfterConfigurationProcess** (**F1.A2)** value shall be located in a logical block that supports software. (see to ref [2])

#### 7.10.1 Read/Write operations

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_010_a**
In all session of applicative software, the ECU shall answer positively to **READ** the **ConfigurationDataReferenceAfterConfigurationProcess** ($22.**F1.A2**), using the dataRecord parameter as defined in [7.10.4].

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_020_a**
In default session of applicative software, the ECU shall answer negatively to **WRITE** the **ConfigurationDataReferenceAfterConfigurationProcess** ($2E.**F1.A2**), with Error code 0x31 (7F.2E.31).

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_025_a**
In extended session of applicative software, the ECU shall answer positively to **WRITE** the **ConfigurationDataReferenceAfterConfigurationProcess** ($2E.**F1.A2**), using the dataRecord parameter as defined in [7.10.4].

*Remark: The OEM tool doesn't read the **ConfigurationDataReferenceAfterConfigurationProcess** DID in bootloader software.*
*Whatever the supplier **ConfigurationDataReferenceAfterConfigurationProcess** implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [7.10.4].*

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_040_a**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** **ConfigurationDataReferenceAfterConfigurationProcess** ($2E.**F1.A2**), with Error code 0x31 (7F.2E.31).

#### 7.10.2 Initialization

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_100_a**
For a flashable ECU in **REFB,** when delivered to OEM, the **ConfigurationDataReferenceAfterConfigurationProcess (F1.A2)** shall contain the **order part reference (REFB)**.

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_110_a**
For a flashable ECU in **REFC,** when delivered to OEM, the **ConfigurationDataReferenceAfterConfigurationProcess (F1.A2)** shall contain the **pdx file reference**.

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_120_a**
For a **non** flashable ECU in **REFC,** when delivered to OEM, the **ConfigurationDataReferenceAfterConfigurationProcess (F1.A2)** shall contain the **order part reference (REFC)**.

### 7.10.3 OEM programming process Behavior

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_200_a**
For an ECU in **REFB,** when delivered to OEM,
**WHEN** the **OEM** programming process download new software,
**THEN** the binary of the software will be downloaded.
The binary of the calibration will be downloaded also **BECAUSE** calibration binary will contain new **pdx file reference** value.
The **ConfigurationDataReferenceAfterConfigurationProcess** DID shall be equal to the new **pdx file reference** depending on this new software and calibration.

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_210_a**
For an ECU in **REFC**, when delivered to OEM and the software **needs** a calibration to run,
**WHEN** the programming process download new software,
**THEN** the binary of the software will be downloaded.
The binary of the calibration will be downloaded also **BECAUSE** calibration binary will contain new **pdx file reference** value.
The **ConfigurationDataReferenceAfterConfigurationProcess** DID shall be equal to the new **pdx file reference** depending on this new software and calibration.

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_220_a**
For an ECU in **REFC**, when delivered to OEM and the software **doesn't need** calibration to run,
**WHEN** the programming process download new software,
**THEN** the binary of the software will be downloaded, software binary will contain new (**Pdx file reference)** value.
The **ConfigurationDataReferenceAfterConfigurationProcess** DID shall be equal to the new **pdx file reference** depending on this new software.

*Remark: The following table explains the life behavior of the ConfigurationDataReferenceAfterConfigurationProcess (F1.A2)*

| Use case | Part Ref | Delivered part | F1.A2 value when delivered by supplier | F1.A2 value, after download only the SW by OEM processing in Plant/after sale | F1.A2 value, after download SW and Calibration by OEM processing in Plant/after sale | F1.A2 value, after configuration processing (write $2E configuration) |
|---|---|---|---|---|---|---|
| 1<br>flashable + configurable | REFB | HW + SW | (REFB) | No effect, but after downloading the SW, the calibration will be download automatically and after that the F1.A2 will be modified automatically<br>F1.A2= pdx file reference | F1.A2 is modified automatically when the binary of the SW AND the binary of the calibration are download.<br>F1.A2= pdx file reference | F1.A2 is modified to the configuration reference written by OEM tool. |
| 2<br>flashable + configurable | REFC | HW+SW+Calibration | pdx file reference | No effect, but after downloading the SW, the calibration will be download automatically. | F1.A2 is modified automatically when the binary of the SW AND the binary of the calibration are download.<br>F1.A2=pdx file reference | F1.A2 is modified to the configuration reference written by OEM tool. |
| 3<br>flashable + configurable | REFC | HW+SW without calibration because not need | pdx file reference | F1.A2 is modified automatically when the binary of the SW is download.<br>F1.A2=pdx file reference | N/A | F1.A2 is modified to the configuration reference written by OEM tool. |
| 4<br>Not-flashable + configurable | REFC | HW+SW+Calibration | (REFC) | N/A | N/A | F1.A2 is modified to the configuration reference written by OEM tool. |
| 5<br>Not-flashable + configurable | REFC | HW+SW without calibration because not need | (REFC) | N/A | N/A | F1.A2 is modified to the configuration reference written by OEM tool. |

**Table 7**: ConfigurationDataReferenceAfterConfigurationProcess (F1.A2) life behavior for OEM

### 7.10.4 dataRecord definition

**RuleReference_ConfigurationDataReferenceAfterConfigurationProcess_300_a**
The ECU shall use the **ConfigurationDataReferenceAfterConfigurationProcess** (**F1.A2**) dataRecord parameter, as defined in the following table.

| Size (bytes) | Format | Coding | Forbidden Character | Data defined by |
|---|---|---|---|---|
| 10 Fix | ASCII | Example:<br>$34$37$39$35$32$30$30$32$33$52 = 479520023R | - | OEM |

# 8 ECU SOFTWARE SECURITY

## 8.1 Digest (FD.00-FE.FF)

Needs: Digest is used by plant tool, DDT2000 and after sale tool to know which logical block shall be downloaded. If an ECU logical block is the same than the pdx file logical block, its digest will be equal and the tool doesn't download it. Digest is also used by the ECU to detect a download error.

**RuleReference_Digest_000_b**
**Digest** shall be implemented ONLY by all flashable ECU designed for ALLIANCE, according to ref[2].

**RuleReference_Digest_005_a**
The flashable ECU shall implement for each logical block, a **Digest** reported by a DataIdentifier. The DataIdentifier shall be in the system supplier specific range [FD.00 to FE.FF].

### 8.1.1 Read/Write operations

**RuleReference_Digest_010_a**
For a flashable ECU, in programming session of bootloader software, the ECU shall answer positively to **READ Digest** ($22.**FD.00** to $22.**FE.FF**)

**RuleReference_Digest_015_a**
For a flashable ECU, in default session of bootloader software, the ECU shall answer positively to **READ Digest** ($22.**FD.00** to $22.**FE.FF**)

**RuleReference_Digest_020_b**
In all sessions of applicative software, the ECU shall answer negatively to **READ Digest** ($22.**FD.00** to $22.**FE.FF**), with Error code 0x31 (7F.22.31).

**RuleReference_Digest_030_b**
In default session of bootloader software, the ECU shall answer negatively to **WRITE Digest** ($2E.**FD.00** to $2E.**FE.FF**), with Error code 0x31 (7F.2E.31).

**RuleReference_Digest_035_a**
In programming session of bootloader software, the ECU shall answer positively to **WRITE Digest** ($2E.**FD.00** to $2E.**FE.FF**) **IF SECURITY_STATE** equals **"unlocked"**.

**RuleReference_Digest_040_a**
In programming session of bootloader software, the ECU shall answer negatively to **WRITE Digest** ($2E.**FD.00** to $2E.**FE.FF**) **IF SECURITY_STATE** equals **"locked"**, with Error code 0x33 (7F.2E.33).

**RuleReference_Digest_055_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE Digest** ($2E.**FD.00** to $2E.**FE.FF**), with Error code 0x31 (7F.2E.31)

### 8.1.2 Initialization

**RuleReference_Digest_100_b**
**Digest** shall be created and managed by the supplier.

**RuleReference_Digest_110_b**
If **Digest** length is less than the max length authorized, according to [8.1.3], the ECU can answer **ONLY** the useful bytes.

**RuleReference_Digest_120_a**
**Digest** length defined in the pdx file shall be equal with the **Digest** length calculated by the Bootloader software.

### 8.1.3 DataRecord definition

**RuleReference_Digest_300_b**
The ECU shall use **Digest** (**FD.00** to **FE.FF**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 32 Max | HEXA | Defined by supplier | - | Supplier |

## 8.2 Fingerprint (F0.10)

Needs: The FingerPrint is used by the ECU using the Security Access for the programming process or write secure DID.

**RuleReference_Fingerprint_000_b**
The **Fingerprint** shall be implemented by all ECU which implement the specification [4]

### 8.2.1 Read/Write operations

**RuleReference_Fingerprint_010_a**
For a flashable ECU, in programming session of bootloader software, the ECU shall answer positively to **READ** the **Fingerprint** ($22.**F0.10**).

**RuleReference_Fingerprint_012_a**
For a flashable ECU, in default session of bootloader software, the ECU shall answer positively to **READ** the **Fingerprint** ($22.**F0.10**).

**RuleReference_Fingerprint_015_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **Fingerprint** ($22.**F0.10**).

**RuleReference_Fingerprint_020_b**
In extended session of applicative software, the ECU shall answer positively to **READ** the **Fingerprint** ($22.**F0.10**).

**RuleReference_Fingerprint_030_a**
In all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **Fingerprint** ($2E.**F0.10**), with Error code 0x31 (7F.2E.31).

**RuleReference_Fingerprint_050_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **Fingerprint** ($2E.**F0.10**), with Error code 0x31 (7F.2E.31)

### 8.2.2 Initialization

**RuleReference_Fingerprint_100_b**
The **Fingerprint (F0.10)** shall be created and sent by the OEM to the ECU during the security access process, according to [8.2.3].
*Origin: Checking SecurityKey in 36-00-030*

**RuleReference_Fingerprint_110_b**
When the part is delivered to OEM, the **Fingerprint** (**F0.10**) DID shall be equal to $20.20.20.20.20.20.20.20.20.20.20.20.20.20.20.20

### 8.2.3 Applicative software and bootloader software Behavior

**RuleReference_Fingerprint_260_a**
If the applicative software modify the **Fingerprint**, the bootloader shall return this new value when it receives the **Fingerprint** ($22.**F0.10**) reading request.

**RuleReference_Fingerprint_265_a**
If the bootloader software modify the **Fingerprint**, the applicative software shall return this new value when it receives the **Fingerprint** ($22.**F0.10**) reading request.

### 8.2.4 DataRecord definition

**RuleReference_Fingerprint_300_b**
The ECU shall use the **Fingerprint** (F**0.10**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 16 Fix | HEXA | - | - | OEM |

### 8.3 IndexSrvData (F0.11)

Needs: The IndexSrvData is one of information which is used by the offboard tools to unlock the ECU.

**RuleReference_IndexSrvData_000_a**
The **IndexSrvData** shall be implemented by all ECU which implement the specification [4].

#### 8.3.1 Read/Write operations

**RuleReference_IndexSrvData_010_a**
For a flashable ECU, in programming session of bootloader software, the ECU shall answer positively to **READ** the **IndexSrvData** ($22.**F0.11**).

**RuleReference_IndexSrvData_020_a**
For a flashable ECU, in default session of bootloader software, the ECU shall answer positively to **READ** the **IndexSrvData** ($22.**F0.11**).

**RuleReference_IndexSrvData_025_a**
If the applicative software implement the Security Access, in extended session of applicative software, the ECU shall answer positively to **READ** the **IndexSrvData** ($22.**F0.11**).

**RuleReference_IndexSrvData_030_a**
If the applicative software implement the Security Access, in default session of applicative software, the ECU shall answer positively to **READ** the **IndexSrvData** ($22.**F0.11**).

**RuleReference_IndexSrvData_040_a**
In all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **IndexSrvData** ($2E.**F0.11**), with Error code 0x31 (7F.2E.31).

**RuleReference_IndexSrvData_050_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **IndexSrvData** ($2E.**F0.11**), with Error code 0x31 (7F.2E.31)

#### 8.3.2 Initialization

**RuleReference_IndexSrvData_100_a**
When the part is delivered to OEM, the **IndexSrvData** (**F0.11**) DID, used by the applicative software and bootloader software, shall be the same for both software, and equal to the information created by OEM and sent to supplier at the beginning of the project.

#### 8.3.3 OEM programming process Behavior

**RuleReference_IndexSrvData_200_b**
Any modification of **IndexSrvData** shall be forbidden by OEM and supplier tools.

#### 8.3.4 dataRecord definition

**RuleReference_IndexSrvData_300_a**
The ECU shall use the **IndexSrvData** (F**0.11**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 10 Max | ASCII | - | - | OEM |

# 9 ECU IDENTIFICATION

## 9.1 BootVersion (F1.80)

This information is to identify the release of the bootloader.

**RuleReference_BootVersion_001_c**
The **BootVersion** (**F1.80**) DID shall be implemented by all ALLIANCE ECU.

### 9.1.1 Read/Write operations

**RuleReference_BootVersion_010_b**
**For a flashable ECU,**
in default session of bootloader software, the ECU shall answer positively to **READ** the **BootVersion** ($22.**F1. 80**), using the dataRecord parameter as defined in [9.1.4]

**RuleReference_BootVersion_015_b**
**For a flashable ECU,**
in programming session of bootloader software, the ECU shall answer positively to **READ** the **BootVersion** ($22.**F1. 80**), using the dataRecord parameter as defined in [9.1.4]

**RuleReference_BootVersion_020_a**
In all sessions of applicative software, the ECU shall answer negatively to **READ** the **BootVersion** ($22.**F1. 80**), with Error code 0x31 (7F.22.31)

**RuleReference_BootVersion_025_b**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **BootVersion** ($2E.**F1. 80**)**,** with Error code 0x31 (7F.2E.31)

**RuleReference_BootVersion_030_b**
In all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **BootVersion** ($2E.**F1. 80**)**,** with Error code 0x31 (7F.2E.31)

### 9.1.2 Initialization

**RuleReference_BootVersion_100_a**
The **BootVersion** information shall be created and managed by the supplier and it shall be changed at each new bootloader software release.

**RuleReference_BootVersion_110_b**
When delivered to OEM, the **BootVersion** (**F1.80**) DID shall be initialized according to [9.1.4].

**RuleReference_BootVersion_120_a**
If the **BootVersion** length is less than the max length authorized, the ECU can answer ONLY the useful bytes.

**RuleReference_BootVersion_130_a**
If the ECU answers to a **BootVersion** read request, using the max length, it shall use ONLY padding bytes with $20.

**RuleReference_BootVersion_140_a**
The first data byte of the **BootVersion (F1.80) DID** is always equals to $01.

### 9.1.3 OEM programming process behavior

**RuleReference_BootVersion_200_a**
When the programming process downloads new software or calibration, the **BootVersion (F1.80)** DID shall never change.

### 9.1.4 dataRecord definition

**RuleReference_BootVersion_300_b**
The ECU shall use the **BootVersion** (**F1.80)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 1 Fix | HEXA | Number of Module: $01 | - | OEM |
| 31 Max | ASCII | Supplier coding.<br>Example: SWBOOT001 => $53 57 42 4F 4F 54 30 30 31 | [$00-$1F] | Supplier |

## 9.2 CalibrationNumber (F1.82)

This information is only used by the ECU designer, if he needs it.

**RuleReference_CalibrationNumber_000_b**
The **CalibrationNumber** (**F1.82**) DID can be implemented by all ALLIANCE ECU, **IF** the ECU designer **AND** the supplier needs to use this information.

### 9.2.1 Read/Write operations

*Remark: The OEM tool doesn't **read** the **CalibrationNumber** DID in bootloader software.*
*Whatever the supplier **CalibrationNumber** implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [9.2.4].*

**RuleReference_CalibrationNumber_010_a**
**IF** the **CalibrationNumber** is implemented**,**
in default session of applicative software, the ECU shall answer positively to **READ** the **CalibrationNumber** ($22.**F1.82**), using the dataRecord parameter as defined in [9.2.4]
**ELSE** it can reply negatively with Error code 0x31 (7F.22.31)

**RuleReference_CalibrationNumber_015_a**
**IF** the **CalibrationNumber** is implemented**,**
in extended session of applicative software, the ECU shall answer positively to **READ** the **CalibrationNumber** ($22.**F1.82**), using the dataRecord parameter as defined in [9.2.4]
**ELSE** it can reply negatively with Error code 0x31 (7F.22.31)

**RuleReference_CalibrationNumber_020_a**
For a flashable ECU, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **CalibrationNumber** ($2E.**F1.82**), with Error code 0x31 (7F.2E.31)

**RuleReference_CalibrationNumber_025_a**
In all sessions of applicative software, the ECU can answer negatively to **WRITE** the **CalibrationNumber** ($2E.**F1.82**)**,** with Error code 0x31 (7F.2E.31)

### 9.2.2 Initialization

**RuleReference_CalibrationNumber_100_a**
The **CalibrationNumber** shall be created and managed by the supplier and it shall be different for each calibration.

**RuleReference_CalibrationNumber_110_b**
When delivered to OEM, If the ECU is not calibrated, the **CalibrationNumber** (**F1.82**) DID shall be equal to $00.00.00.00.00.00.00.00.00.00.

**RuleReference_CalibrationNumber_120_b**
When delivered to OEM, If the ECU is already calibrated, the **CalibrationNumber** (**F1.82**) DID shall be initialized according to [9.2.4].

**RuleReference_CalibrationNumber_130_a**
If the **CalibrationNumber** length is less than the max length authorized, the ECU can answer ONLY the useful bytes.

**RuleReference_CalibrationNumber_140_b**
If the ECU answers to a **CalibrationNumber** read request, using the max length, it shall use ONLY padding bytes with $00.

**RuleReference_CalibrationNumber_150_a**
The first data byte of the **CalibrationNumber (F1.82) DID** is always equals to $01.

### 9.2.3 Behavior after OEM programming process

**RuleReference_CalibrationNumber_200_a**
When the programming process download new calibration, the **CalibrationNumber (F1.82)** DID shall be changed according with this new tuning.

### 9.2.4 dataRecord definition

**RuleReference_CalibrationNumber_300_b**
The ECU shall use the **CalibrationNumber** (**F1.82)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 1 Fix | HEXA | Number of module: $01 | - | OEM |
| 9 Max | HEXA | Supplier coding.<br>Example: 347800101213141516 => $34 78 00 10 12 13 14 15 16 | - | Supplier |

### 9.3 systemSupplierIdentifier (F1.8A)

The systemSupplierIdentifier is used to reference the system supplier name.

**RuleReference_systemSupplierIdentifier_000_b**
The **systemSupplierIdentifier** (**F1.8A**) shall be implemented by all ECU designed for ALLIANCE.

#### 9.3.1 Read/Write operations

**RuleReference_systemSupplierIdentifier_010_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **systemSupplierIdentifier** ($22.**F1.8A**), using the dataRecord parameter as defined in [9.3.4].

**RuleReference_systemSupplierIdentifier_020_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **systemSupplierIdentifier** ($22.**F1.8A**), using the dataRecord parameter as defined in [9.3.4].

**RuleReference_systemSupplierIdentifier_022_b**
If the ECU is flashable, in default session of bootloader software, the ECU shall answer positively to **READ** the **systemSupplierIdentifier** ($22.**F1.8A**), using the dataRecord parameter as defined in [9.3.4].

**RuleReference_systemSupplierIdentifier_023_b**
If the ECU is flashable, in programming session of bootloader software, the ECU shall answer positively to **READ** the **systemSupplierIdentifier** ($22.**F1.8A**), using the dataRecord parameter as defined in [9.3.4].

**RuleReference_systemSupplierIdentifier_025_a**
If the ECU is flashable, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** **systemSupplierIdentifier** ($2E.**F1.8A**), with Error code 0x31.

**RuleReference_systemSupplierIdentifier_030_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **systemSupplierIdentifier** ($2E.**F1.8A**), with Error code 0x31.

#### 9.3.2 Initialization

**RuleReference_systemSupplierIdentifier_100_b**
When delivered to OEM, the **systemSupplierIdentifier** (**F1.8A**) DID shall be initialized according to [9.3.4].

**RuleReference_systemSupplierIdentifier_110_a**
If the **systemSupplierIdentifier** length is less than the max length authorized, the ECU can answer ONLY the useful bytes.

**RuleReference_systemSupplierIdentifier_120_a**
If the ECU answers to a **systemSupplierIdentifier** read request, using the max length, it shall use ONLY padding bytes with $20.

### 9.3.3 OEM programming process behavior

**RuleReference_systemSupplierIdentifier_200_a**
When the programming process downloads new software or calibration, the **systemSupplierIdentifier (F1.8A)** DID shall never change.

### 9.3.4 dataRecord definition

**RuleReference_systemSupplierIdentifier_300_a**
The ECU shall use the **systemSupplierIdentifier** (**F1.8A**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 64 Max | UTF-8 | Example:<br>Renault = $52$65$6E$61$75$4C$74<br>22 F1.8A => 62 F1 8A 52 65 6E 61 75 4C 74 | [$00-$1F] | SUPPLIER |

### 9.4 SystemSupplierECUsoftwareNumber (F1.94)

Needs: This information is used by the engineering tool to identify the ECU.

**RuleReference_SystemSupplierECUsoftwareNumber_000_b**
The **SystemSupplierECUsoftwareNumber** (**F1.94**) shall be implemented by all ECU designed for ALLIANCE.

#### 9.4.1 Read/Write operations

**RuleReference_SystemSupplierECUsoftwareNumber_010_a**
In default session of applicative software, the ECU shall answer positively to **READ SystemSupplierECUsoftwareNumber** ($22.**F1.94**), using the dataRecord parameter as defined in [9.4.4].

**RuleReference_SystemSupplierECUsoftwareNumber_015_a**
In extended session of applicative software, the ECU shall answer positively to **READ SystemSupplierECUsoftwareNumber** ($22.**F1.94**), using the dataRecord parameter as defined in [9.4.4].

*Remark: The OEM tool doesn't read the **SystemSupplierECUsoftwareNumber** DID in bootloader software. Whatever the supplier **SystemSupplierECUsoftwareNumber** implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [9.4.4].*

**RuleReference_SystemSupplierECUsoftwareNumber_020_a**
If the ECU is flashable, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE SystemSupplierECUsoftwareNumber** ($2E.**F1.94**), with Error code 0x31 (7F.2E.31).

**RuleReference_SystemSupplierECUsoftwareNumber_025_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE SystemSupplierECUsoftwareNumber** ($2E.**F1.94**), with Error code 0x31 (7F.2E.31).

#### 9.4.2 Initialization

**RuleReference_SystemSupplierECUsoftwareNumber_100_a**
The **SystemSupplierECUsoftwareNumber** shall be created and managed by the supplier and it shall be changed at each new applicative software release.

**RuleReference_SystemSupplierECUsoftwareNumber_110_b**
When delivered to OEM, the **SystemSupplierECUsoftwareNumber** (**F1.94**) DID shall be initialized according to [9.4.4].

**RuleReference_SystemSupplierECUsoftwareNumber_120_a**
If the **SystemSupplierECUsoftwareNumber (F1.94)** length is less than the max length authorized, the ECU can answer ONLY the useful bytes.

**RuleReference_SystemSupplierECUsoftwareNumber_130_a**
If the ECU answers to a **SystemSupplierECUsoftwareNumber (F1.94)** read request, using the max length, it shall use ONLY padding bytes with $20.

### 9.4.3 OEM programming process behavior

**RuleReference_SystemSupplierECUsoftwareNumber_200_a**
When the programming process downloads new software, the **SystemSupplierECUsoftwareNumber (F1.94)** DID shall be change.

**RuleReference_SystemSupplierECUsoftwareNumber_210_a**
When the programming process downloads ONLY a new calibration, the **SystemSupplierECUsoftwareNumber (F1.94)** DID shall never change.

### 9.4.4 dataRecord definition

**RuleReference_SystemSupplierECUsoftwareNumber_300_a**
The ECU shall use the **SystemSupplierECUsoftwareNumber** (**F1.94)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 32 Max | UTF8 | Supplier coding<br>Example: SW01456 => $62 F1 94 53 30 31 34 35 36 | [$00-$1F] | Supplier |

### 9.5 SystemSupplierECUSoftwareVersionNumber (F1.95)

Needs: This information is used by the engineering tool to identify the ECU.

**RuleReference_SystemSupplierECUSoftwareVersionNumber_000_b**
The **SystemSupplierECUSoftwareVersionNumber** (**F1.95**) shall be implemented by all ECU designed for ALLIANCE.

#### 9.5.1 Read/Write operations

**RuleReference_SystemSupplierECUSoftwareVersionNumber_010_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **SystemSupplierECUSoftwareVersionNumber** ($22.**F1.95**), using the dataRecord parameter as defined in [9.5.4].

**RuleReference_SystemSupplierECUSoftwareVersionNumber_015_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **SystemSupplierECUSoftwareVersionNumber** ($22.**F1.95**), using the dataRecord parameter as defined in [9.5.4].

*Remark: The OEM tool doesn't read the **SystemSupplierECUSoftwareVersionNumber** DID in bootloader software.*
*Whatever the supplier **SystemSupplierECUSoftwareVersionNumber** implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [9.5.4].*

**RuleReference_SystemSupplierECUSoftwareVersionNumber_020_a**
If the ECU is flashable, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **SystemSupplierECUSoftwareVersionNumber** ($2E.**F1.95**), with Error code 0x31 (7F.2E.31).

**RuleReference_SystemSupplierECUSoftwareVersionNumber_025_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **SystemSupplierECUSoftwareVersionNumber** ($2E.**F1.95**)**,** with Error code 0x31 (7F.2E.31).

#### 9.5.2 Initialization

**RuleReference_SystemSupplierECUSoftwareVersionNumber_100_a**
The **SystemSupplierECUSoftwareVersionNumber** shall be created and managed by the supplier and it shall be changed at each new applicative software release.

**RuleReference_SystemSupplierECUSoftwareVersionNumber_110_b**
When delivered to OEM, the **SystemSupplierECUSoftwareVersionNumber** (**F1.95**) DID shall be initialized according to [9.5.4].

**RuleReference_SystemSupplierECUSoftwareVersionNumber_120_a**
If the **SystemSupplierECUSoftwareVersionNumber (F1.95)** length is less than the max length authorized, the ECU can answer ONLY the useful bytes.

**RuleReference_SystemSupplierECUSoftwareVersionNumber_130_a**
If the ECU answers to a **SystemSupplierECUSoftwareVersionNumber (F1.95)** read request, using the max length, it shall use ONLY padding bytes with $20.

#### 9.5.3 OEM programming process behavior

**RuleReference_SystemSupplierECUSoftwareVersionNumber_200_b**
When the programming process downloads new software, the **SystemSupplierECUSoftwareVersionNumber (F1.95)** DID shall be change, in accordance with the corresponding value of the new software release, defined by the supplier.
**RuleReference_SystemSupplierECUSoftwareVersionNumber_210_a**
When the programming process downloads ONLY a new calibration, the **SystemSupplierECUSoftwareVersionNumber (F1.95)** DID shall never change.

### 9.5.4 dataRecord definition

**RuleReference_SystemSupplierECUSoftwareVersionNumber_300_a**
The ECU shall use the **SystemSupplierECUSoftwareVersionNumber** (**F1.95)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 32 Max | UTF8 | Supplier coding.<br>Example: REL2013JAN04 => $62 F1 95 52 45 4C 32 30 31 33 4A 41 4E 30 34 | [$00-$1F] | Supplier |

### 9.6 VDIAG (F1.A0)

Needs: To identify the right repair method, the OEM defines for each ECU a specific VDIAG. This information shall be done by OEM diagnostic team.

**RuleReference_VDIAG_000_b**
The **VDIAG** (**F1.A0**) shall be implemented by all ECU designed for ALLIANCE.

#### 9.6.1 Read/Write operations

**RuleReference_VDIAG_010_a**
In all sessions of applicative software, the ECU shall answer positively to **READ** the **VDIAG** ($22.**F1.A0**), using the dataRecord parameter as defined in [9.6.2].

**RuleReference_VDIAG_015_a**
In all sessions of bootloader software, the ECU shall answer negatively to READ the VDIAG ($22.F1.A0), with Error code 0x31 (7F.22.31).
Origin: C3+ needs to have this information in bootloader to start the reprogramming process.

**RuleReference_VDIAG_020_a**
If the ECU is flashable, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE VDIAG** ($2E.**F1.A0**), with Error code 0x31 (7F.2E.31).

**RuleReference_VDIAG_025_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **VDIAG** ($2E.**F1.A0**), with Error code 0x31 (7F.2E.31).

*Remark: The **VDIAG** is stored in the software. If the OEM decides to change it, the new **VDIAG** will be initialized in the new software and it will be take into account in the pdx file.*

#### 9.6.2 Initialization

**RuleReference_VDIAG_100_b**
When delivered to OEM, the **VDIAG** (**F1.A0**) DID shall be initialized according to [9.6.4].

#### 9.6.3 OEM programming process behavior

**RuleReference_VDIAG_200_a**
When the programming process downloads new software, the **VDIAG (F1.A0)** DID shall be change ONLY if the ECU designer, in accordance with diagnostic department, want to change it.

**RuleReference_VDIAG_210_a**
When the programming process downloads a new calibration, the **VDIAG (F1.A0)** DID shall never change.

#### 9.6.4 dataRecord definition

**RuleReference_VDIAG_300_b**
The ECU shall use the **VDIAG** (**F1.A0)** dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 1 Fix | HEX | Defined by OEM<br>Example: 14 => $62 F1 A0 14 | - | OEM |

# 10 TRACEABILITY

## 10.1 ECUSerialNumberDataIdentifier(F1.8C)

Remarks: The ECU serial number is used for quality department in Renault and Nissan side.
Furthermore, it is also used for functional safety department in Nissan side.

**RuleReference_ECUSerialNumberDataIdentifier_000_b**
The **ECUSerialNumberDataIdentifier** (**F1.8C**) shall be implemented by all ECU designed for ALLIANCE.

### 10.1.1 Read/Write operations

**RuleReference_ECUSerialNumberDataIdentifier_010_b**
For a flashable ECU, in default session of bootloader software, the ECU shall answer positively to **READ** the **ECUSerialNumberDataIdentifier** ($22.**F1.8C**).

**RuleReference_ECUSerialNumberDataIdentifier_012_a**
For a flashable ECU, in programming session of bootloader software, the ECU shall answer positively to **READ** the **ECUSerialNumberDataIdentifier** ($22.**F1.8C**).

**RuleReference_ECUSerialNumberDataIdentifier_015_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **ECUSerialNumberDataIdentifier** ($22.**F1.8C**), using the dataRecord parameter as defined in [10.1.4].

**RuleReference_ECUSerialNumberDataIdentifier_020_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **ECUSerialNumberDataIdentifier** ($22.**F1.8C**), using the dataRecord parameter as defined in [10.1.4].

**RuleReference_ECUSerialNumberDataIdentifier_025_a**
If the ECU is flashable, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE** **ECUSerialNumberDataIdentifier** ($2E.**F1.8C**), with Error code 0x31 (7F.2E.31).

**RuleReference_ECUSerialNumberDataIdentifier_030_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **ECUSerialNumberDataIdentifier** ($2E.**F1.8C**), with Error code 0x31 (7F.2E.31).

### 10.1.2 Initialization

**RuleReference_ECUSerialNumberDataIdentifier_100_b**
The **ECUSerialNumberDataIdentifier** (**F1.8C**) shall

- be managed by the supplier **AND**
- be unique for each part

*Remark: The supplier shall give to OEM the algorithm which explain and demonstrate that each ECU will have a unique serial number.*

**RuleReference_ECUSerialNumberDataIdentifier_110_b**
When delivered to OEM, the **ECUSerialNumberDataIdentifier** (**F1.8C**) DID shall be initialized according to [10.1.4].

**RuleReference_ECUSerialNumberDataIdentifier_140_a**
If the **ECUSerialNumberDataIdentifier** (**F1.8C**) DID length is less than max value, the ECU can answer ONLY the useful bytes.

**RuleReference_ECUSerialNumberDataIdentifier_130_a**
If the ECU answers to an **ECUSerialNumberDataIdentifier (F1.8C)** read request, using the max length, it shall use **ONLY** padding bytes with $20.

### 10.1.3 OEM programming process behavior

**RuleReference_ECUSerialNumberDataIdentifier_200_a**
The **ECUSerialNumberDataIdentifier** (**F1.8C**) shall never change whatever the process used.

### 10.1.4 dataRecord definition

**RuleReference_ECUSerialNumberDataIdentifier_300_b**
The ECU shall use the **ECUSerialNumberDataIdentifier**(**F1.8C**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 20 Fix | ASCII | In the ranges A to Z, a to z, 0 to 9.<br>The space characters, “$20”, will be used for padding. | [$00-$1F] | ~~Renault ECU:~~ Supplier<br>~~Nissan ECU: Nissan~~ |

### 10.2 exhaustRegulationOrTypeApprovalNumberDataIdentifier (F1.96)

**RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_000_b**
The **exhaustRegulationOrTypeApprovalNumberDataIdentifier** shall be implemented by all **OBD** ECU designed for ALLIANCE.

#### 10.2.1 Read/Write operations

**RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_010_a**
**IF** the ECU is **OBD**,
in default session of applicative software, it shall answer positively to **READ** the **exhaustRegulationOrTypeApprovalNumberDataIdentifier** ($22.**F1.96**), using the dataRecord parameter as defined in [10.2.2],
**Else** the non-OBD ECU answers negatively with the Error code 0x31 (7F.22.31).

**RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_015_c**
**IF** the ECU is **OBD**,
in extended session of applicative software, the ECU shall answer positively to **READ** the **exhaustRegulationOrTypeApprovalNumberDataIdentifier** ($22.**F1.96**), using the dataRecord parameter as defined in [10.2.2],
**Else** the non-OBD ECU answers negatively with the Error code 0x31 (7F.22.31).

*Remark: The OEM tool doesn't read the **exhaustRegulationOrTypeApprovalNumberDataIdentifier** DID in bootloader software.*
*Whatever the supplier **exhaustRegulationOrTypeApprovalNumberDataIdentifier** implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [10.2.4].*

**RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_025_a**
In all sessions of bootloader software, the ECU shall answer negatively to **WRITE** **exhaustRegulationOrTypeApprovalNumberDataIdentifier** ($2E.**F1.96**), with Error code 0x3 (7F.2E.31).

**RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_030_a**
In all sessions of applicative software, the OBD ECU shall answer negatively to **WRITE** the **exhaustRegulationOrTypeApprovalNumberDataIdentifier** ($2E.**F1.96**), with Error code 0x31 (7F.2E.31).

#### 10.2.2 Initialization

**RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_100_b**
When delivered to OEM, the **exhaustRegulationOrTypeApprovalNumberDataIdentifier** (**F1.96**) DID shall be initialized according to [10.2.4].

#### 10.2.3 OEM programming process behavior

**RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_200_a**
The **exhaustRegulationOrTypeApprovalNumberDataIdentifier** (**F1.96**) shall never change whatever the process used.

### 10.2.4 dataRecord definition

**RuleReference_exhaustRegulationOrTypeApprovalNumberDataIdentifier_300_a**
The OBD ECU shall use the **exhaustRegulationOrTypeApprovalNumberDataIdentifier** (**F1.96**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 10 Fix | ASCII | - | [$00-$1F] | OEM |

### 10.3 SystemNameOrEngineType (F1.97)

The SystemNameOrEnginType is used to reference the system name or engine type. This DID is only used for OBD ECU.

**RuleReference_SystemNameOrEngineType_000_b**
The **SystemNameOrEngineType** shall be implemented by all OBD ECU designed for ALLIANCE.

#### 10.3.1 Read/Write operations

**RuleReference_SystemNameOrEngineType_015_a**
**IF** the ECU is **OBD**,
In all sessions of applicative software, it shall answer positively to **READ** the **SystemNameOrEngineType** ($22.**F1.97**), using the dataRecord parameter as defined in [10.3.2].
**Else** the non-OBD ECU answers negatively with the Error code 0x31 (7F.22.31).

*Remark: The OEM tool doesn't read the **SystemNameOrEngineType** DID in bootloader software. Whatever the supplier **SystemNameOrEngineType** implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [10.3.3].*

**RuleReference_SystemNameOrEngineType_025_a**
In all sessions of bootloader software, the ECU shall answer negatively to **WRITE** the **SystemNameOrEngineType** ($2E.**F1.97**), with Error code 0x31 (7F.2E.31).

**RuleReference_SystemNameOrEngineType_030_a**
In all sessions of applicative software, the ECU shall answer negatively to **WRITE** the **SystemNameOrEngineType** ($2E.**F1.97**), with Error code 0x31 (7F.2E.31).

**RuleReference_SystemNameOrEngineType_045_a**
The **SystemNameOrEngineType** (**F1.97**) shall never change whatever the process used.

#### 10.3.2 Initialization

**RuleReference_SystemNameOrEngineType_100_b**
When delivered to OEM, the **SystemNameOrEngineType** (**F1.97**) DID shall be initialized according to [10.3.3].

**RuleReference_SystemNameOrEngineType_110_a**
If the **SystemNameOrEngineType** (**F1.97**) DID length is less than max value, the ECU can answer **ONLY** the useful bytes.

**RuleReference_SystemNameOrEngineType_120_a**
If the ECU answers to an **SystemNameOrEngineType** read request, using the max length, it shall use **ONLY** padding bytes with $20.

### 10.3.3 dataRecord definition

**RuleReference_SystemNameOrEngineType_050_b**
The ECU shall use the **SystemNameOrEngineType** (**F1.97**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Forbidden Character* | *Data defined by* |
|---|---|---|---|---|
| 32 Max | UTF-8 | - | [$00-$1F] | OEM |

### 10.4 VIN (F1.90)

To be compliant with antitheft needs and regulation, “All the ECUs (except immobilizer ECUs) must have no possibility to erase the unitary identification on memory and/or to rewrite another malicious one”, to write again the VIN, SecurityAccess service will be used.

**RuleReference_VIN_005_b**
The **VIN** shall be implemented by all ECU in accordance with Renault/Nissan diagnostic department.

#### 10.4.1 Read/Write operations

*Remark: The OEM tool doesn’t read the **VIN** DID in bootloader software.*
*Whatever the supplier **VIN** implementation in all sessions of bootloader software, it can reply using the DataRecord parameter defined in [10.4.4].*

**RuleReference_VIN_015_a**
In default session of applicative software, the ECU shall answer positively to **READ** the **VIN** ($22.**F1.90**) whatever the VIN value, using the dataRecord parameter as defined in [10.4.4].

*Remark: The ECU shall always answer positively to the read VIN request whatever the VIN value even if the VIN is not initialized.*

**RuleReference_VIN_020_a**
In extended session of applicative software, the ECU shall answer positively to **READ** the **VIN** ($22.**F1.90**) whatever the VIN, using the dataRecord parameter as defined in [10.4.4].

**RuleReference_VIN_030_b**
If the ECU is flashable, in all sessions of bootloader software, the ECU shall answer negatively to **WRITE VIN** ($2E.**F1.90**), with Error code 0x31 (7F.2E.31).

**RuleReference_VIN_040_b**
In default session of applicative software, the ECU shall answer negatively to **WRITE VIN** ($2E.**F1.90**), with Error code 0x31 (7F.2E.31).

**RuleReference_VIN_050_b**
In extended session of applicative software, the ECU shall answer positively to **WRITE** the **VIN** ($2E.**F1.90**)
**IF** the VIN is empty
**AND** (**SECURITY_STATE** equals **“unlocked” OR SECURITY_STATE** equals **“locked”)**.

*Remark:*
- *The parameter **SECURITY_STATE** is defined in the ref [3].*
- *The VIN is empty when all byte equals 0x00 or 0xFF or 0x30*

**RuleReference_VIN_060_b**
The ECU shall answer negatively to a **WRITE VIN** request **WHEN** the value sent by the $2E service is equals to
17 * $00 **OR**
17 * $FF **OR**
17 * $30, with Error code 0x31 (7F.2E.31).

**RuleReference_VIN_070_a**
In extended session of applicative software, the ECU shall answer positively to **WRITE** the **VIN** ($2E.**F1.90**),
**IF** the VIN is **NOT** empty
**AND SECURITY_STATE equals “unlocked”**.

**RuleReference_VIN_080_a**
In extended session of applicative software, the ECU shall answer negatively to **WRITE** the **VIN** ($2E.**F1.90**),
**IF** the VIN is **NOT** empty
**AND SECURITY_STATE equals** “**locked”** with Error code 0x33 (7F.2E.33).

### 10.4.2 Initialization

**RuleReference_VIN_100_b**
To be considered as empty, When delivered to OEM, the **VIN** shall be initialized to 17*$00 or 17*$30 or 17*$FF.

### 10.4.3 OEM programming process behavior

**RuleReference_VIN_200_a**
When the programming process downloads ONLY new software or a new calibration, the **VIN (F1.90)** DID shall never change.

### 10.4.4 dataRecord definition

**RuleReference_VIN_300_b**
The ECU shall use the **VIN** (**F1.90**) dataRecord parameter, as defined in the following table.

| *Size (bytes)* | *Format* | *Coding* | *Data defined by* |
|---|---|---|---|
| 17 (fixed) | ASCII | Example:<br>Value after OEM processing:<br>$56$46$31$32$52$42$44$31$48$34$39$37$37$36$31$35$30 = VF12RBD1H49776150<br>Initial value (empty value):<br>$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30$30 = “00000000000000000”<br>Or<br>$00$00$00$00$00$00$00$00$00$00$00$00$00$00$00$00$00<br>OR<br>$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF$FF | OEM |

# 11 SNAPSHOT INFORMATIONS

## 11.1 Mileage of the last occurrence counter (F0.D0)

Offboard tools needs to have the mileage for each DTC. When a fault appears, the Mileage must be memorized or updated and store in the DTC snapshot associated.

**RuleReference_MILEAGE_LAST_OCCURENCE_005_a**
The **MILEAGE_LAST_OCCURENCE** shall be implemented in all snapshot AND in all DTC, by all ALLIANCE ECU.

### 11.1.1 Read/Write operations

*Remark: The OEM tool doesn't read the **MILEAGE_LAST_OCCURENCE** DID with the service 22, in bootloader software and in applicative Software. The Tool use only the service $1904 to read this information in applicative software.*
*Whatever the supplier **MILEAGE_LAST_OCCURENCE** implementation, if it wants to use the service $22 to read this information, it can reply using the DataRecord parameter defined in [11.1.3].*

**RuleReference_MILEAGE_LAST_OCCURENCE_030_a**
If the ECU is flashable, in **default session** of bootloader software, the ECU shall answer negatively to **WRITE** the **MILEAGE_LAST_OCCURENCE** ($2E.**F0.D0**), with Error code 0x31 (7F.2E.31).

**RuleReference_MILEAGE_LAST_OCCURENCE_035_a**
If the ECU is flashable, in **programming session** of bootloader software, the ECU shall answer negatively to **WRITE** the **MILEAGE_LAST_OCCURENCE** ($2E.**F0.D0**), with Error code 0x31 (7F.2E.31).

**RuleReference_MILEAGE_LAST_OCCURENCE_040_a**
In **default session** of applicative software, the ECU shall answer negatively to **WRITE** the **MILEAGE_LAST_OCCURENCE** ($2E.**F0.D0**), with Error code 0x31 (7F.2E.31).

**RuleReference_MILEAGE_LAST_OCCURENCE_045_a**
In **extended session** of applicative software, the ECU shall answer negatively to **WRITE** the **MILEAGE_LAST_OCCURENCE** ($2E.**F0.D0**), with Error code 0x31 (7F.2E.31).

**RuleReference_MILEAGE_LAST_OCCURENCE_050_a**
The ECU shall support the **MILEAGE_LAST_OCCURENCE DID** (**F0.D0**), only with the request $19.04 "reportDTCSnapshotRecordByDTCNumber" using the dataRecord parameter as defined in [11.1.3].

### 11.1.2 Behavior

**RuleReference_MILEAGE_LAST_OCCURENCE_100_b**
For each DTC, at the first occurrence failure, the ECU shall memorize the mileage corresponding to the current vehicle odometer.

*Remark: When a fault appears, a DTC, with its snapshot, is memorized when the bit0 (TestFailed) and the bit3(ConfirmedDTC) equals 1.*

**RuleReference_MILEAGE_LAST_OCCURENCE_110_b**
For each DTC, the ECU shall update the **MILEAGE_LAST_OCCURENCE** with the current vehicle odometer **ONLY** when the parameter **DTCOccurenceCounter** is increased by one.

**RuleReference_MILEAGE_LAST_OCCURENCE_120_b**
If the current vehicle odometer is not available, the ECU shall initialize the **MILEAGE_LAST_OCCURENCE** with the unavailable value ($FF.FF.FF) **ONLY** for a first occurrence.

**RuleReference_MILEAGE_LAST_OCCURENCE_130_b**
If the current vehicle odometer is not available, the ECU shall NOT initialize the **MILEAGE_LAST_OCCURENCE** with the unavailable value ($FF.FF.FF), if a first occurrence failure was already memorized.

*Remark: When a ClearDiagInfo (14.FF.FF.FF) is send by tools, the ECU shall delete all DTC with all snapshot associated.*

### 11.1.3 dataRecord definition

**RuleReference_MILIEAGE_300_b**
The ECU shall use the **MILEAGE_LAST_OCCURENCE** (**F0.D0**) dataRecord parameter, as defined in the following table.

| *Size (bits)* | *Unit* | *Resolution* | *Offset* | *Min.* | *Max.* | *Coding* |
|---|---|---|---|---|---|---|
| 24 | Km | 1 | 0 | 0 | 999999 km | 0km = $00.00.00<br>999999 km = $0F.42 3F<br>Unavailable value = $FF.FF.FF |

### 11.2 DTC occurrence Counter (DID $F0.D1)

The ECU shall provide for each DTC managed by applicative software, the number of occurrence failure which has been detected.

**RuleReference_DTCOccurrenceCounter_010_A**
The **DTCOccurrenceCounter** shall be implemented in all snapshot **AND** in all DTC, by all ALLIANCE ECU.

#### 11.2.1 Read/Write operations

*Remark: The OEM tool doesn't read the **DTCOccurrenceCounter** DID with the service 22, in bootloader software and in applicative Software. The Tool use only the service $1904 to read this information in applicative software.*
*Whatever the supplier **DTCOccurrenceCounter** implementation, if it wants to use the service $22 to read this information, it can reply using the DataRecord parameter defined in [11.2.3].*

**RuleReference_DTCOccurrenceCounter_040_a**
In **default session** of applicative software, the ECU shall answer negatively to **WRITE** the **DTCOccurrenceCounter** ($2E.**F0.D1**), with Error code 0x31 (7F.2E.31).

**RuleReference_DTCOccurrenceCounter_045_a**
In **extended session** of applicative software, the ECU shall answer negatively to **WRITE** the **DTCOccurrenceCounter** ($2E.**F0.D1**), with Error code 0x31 (7F.2E.31).

**RuleReference_DTCOccurrenceCounter_050_a**
In **default session** of bootloader software, the ECU shall answer negatively to **WRITE** the **DTCOccurrenceCounter** ($2E.**F0.D1**), with Error code 0x31 (7F.2E.31).

**RuleReference_DTCOccurrenceCounter_055_a**
In **programming session** of bootloader software, the ECU shall answer negatively to **WRITE** the **DTCOccurrenceCounter** ($2E.**F0.D1**), with Error code 0x31 (7F.2E.31).

**RuleReference_DTCOccurrenceCounter_060_a**
The ECU shall support the **DTCOccurrenceCounter DID** (**F0.D1**), only with the request $19.04 "reportDTCSnapshotRecordByDTCNumber" using the dataRecord parameter as defined in [11.2.3].

### 11.2.2 Behavior

**RuleReference_DTCOccurenceCounter_110_a**
The initial value of the **DTCOccurrenceCounter** is 0.

**RuleReference_DTCOccurrenceCounter_120_b**
The ECU shall freeze the **DTCOccurrenceCounter** when the maximum value is reached**.**

**RuleReference_DTCOccurenceCounter_130_a**
The ECU shall increase by 1 the **DTCOccurrenceCounter WHEN** a failure is detected and confirmed by the monitoring associated.

**RuleReference_DTCOccurenceCounter_140_A**
**When** a failure is present at end of the ECU electrical cycle **AND IF** at the beginning of the next ECU electrical cycle the failure is still present, the ECU shall not increase the **DTCOccurrenceCounter**.

**RuleReference_DTCOccurenceCounter_150_A**
The ECU shall reset to 0 the **DTCOccurrenceCounter** when it receives the **ClearDiagInfo** request.

### 11.2.3 dataRecord definition

**RuleReference_DTCOccurenceCounter_200_a**
The ECU shall use **DTCOccurrenceCounter** parameter as defined in the following table.

| *Size (bits)* | *Unit* | *Resolution* | *Offset* | *Min.* | *Max. value* | *Coding* | *Meaning* |
|---|---|---|---|---|---|---|---|
| 8 | - | 1 | - | 0 | 255 | - | - |